pdf_code_Unsupervised_Learning_F1_data

May 26, 2025

1 SML HW4 : Formula 1 World Championship Analysis Using Unsupervised Learning

1.0.1 Team Members: Alekhiya, Devi and Hrishabh

1.1 Question 1: Can we group F1 drivers by their race-day performance profiles across seasons using data from results, qualifying, lap times and pit stops?

[138]:
```
%pip install yellowbrick
%pip install plotly
%pip install tabulate
```

```
Defaulting to user installation because normal site-packages is not
writeableNote: you may need to restart the kernel to use updated packages.

Requirement already satisfied: yellowbrick in
c:\users\hrishabh\appdata\roaming\python\python312\site-packages (1.5)
Requirement already satisfied: matplotlib!=3.0.0,>=2.0.2 in
c:\programdata\anaconda3\lib\site-packages (from yellowbrick) (3.8.4)
Requirement already satisfied: scipy>=1.0.0 in
c:\users\hrishabh\appdata\roaming\python\python312\site-packages (from
yellowbrick) (1.11.4)
Requirement already satisfied: scikit-learn>=1.0.0 in
c:\users\hrishabh\appdata\roaming\python\python312\site-packages (from
yellowbrick) (1.5.0)
Requirement already satisfied: numpy>=1.16.0 in
c:\programdata\anaconda3\lib\site-packages (from yellowbrick) (1.26.4)
Requirement already satisfied: cycler>=0.10.0 in
c:\programdata\anaconda3\lib\site-packages (from yellowbrick) (0.11.0)
Requirement already satisfied: contourpy>=1.0.1 in
c:\programdata\anaconda3\lib\site-packages (from
matplotlib!=3.0.0,>=2.0.2->yellowbrick) (1.2.0)
Requirement already satisfied: fonttools>=4.22.0 in
c:\programdata\anaconda3\lib\site-packages (from
matplotlib!=3.0.0,>=2.0.2->yellowbrick) (4.51.0)
Requirement already satisfied: kiwisolver>=1.3.1 in
c:\programdata\anaconda3\lib\site-packages (from
matplotlib!=3.0.0,>=2.0.2->yellowbrick) (1.4.4)
Requirement already satisfied: packaging>=20.0 in
```

```
c:\programdata\anaconda3\lib\site-packages (from
matplotlib!=3.0.0,>=2.0.2->yellowbrick) (23.2)
Requirement already satisfied: pillow>=8 in c:\programdata\anaconda3\lib\site-
packages (from matplotlib!=3.0.0,>=2.0.2->yellowbrick) (10.3.0)
Requirement already satisfied: pyparsing>=2.3.1 in
c:\programdata\anaconda3\lib\site-packages (from
matplotlib!=3.0.0,>=2.0.2->yellowbrick) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in
c:\programdata\anaconda3\lib\site-packages (from
matplotlib!=3.0.0,>=2.0.2->yellowbrick) (2.9.0.post0)
Requirement already satisfied: joblib>=1.2.0 in
c:\programdata\anaconda3\lib\site-packages (from scikit-
learn>=1.0.0->yellowbrick) (1.4.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in
c:\users\hrishabh\appdata\roaming\python\python312\site-packages (from scikit-
learn>=1.0.0->yellowbrick) (3.6.0)
Requirement already satisfied: six>=1.5 in c:\programdata\anaconda3\lib\site-
packages (from python-dateutil>=2.7->matplotlib!=3.0.0,>=2.0.2->yellowbrick)
(1.16.0)
Defaulting to user installation because normal site-packages is not
writeableNote: you may need to restart the kernel to use updated packages.

Requirement already satisfied: plotly in c:\programdata\anaconda3\lib\site-
packages (5.22.0)
Requirement already satisfied: tenacity>=6.2.0 in
c:\programdata\anaconda3\lib\site-packages (from plotly) (8.2.2)
Requirement already satisfied: packaging in c:\programdata\anaconda3\lib\site-
packages (from plotly) (23.2)
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: tabulate in c:\programdata\anaconda3\lib\site-
packages (0.9.0)
Note: you may need to restart the kernel to use updated packages.
```

[139]:
```
import re
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.preprocessing import LabelEncoder, StandardScaler
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA
from sklearn.cluster import KMeans
from yellowbrick.cluster import KElbowVisualizer, SilhouetteVisualizer
from sklearn.metrics import silhouette_score
from collections import Counter
import plotly.graph_objects as go
from matplotlib.colors import LinearSegmentedColormap
```

```
from matplotlib import colors as mcolors
from tabulate import tabulate
from scipy.cluster.hierarchy import linkage, dendrogram, fcluster
```

1.1.1 Load Data

[140]:
```
dataframes = {
    name: pd.read_csv(f"{name}.csv") for name in [
        "circuits", "constructor_results", "constructor_standings",␣
 ↪"constructors",
        "driver_standings", "drivers", "lap_times", "pit_stops",
        "qualifying", "races", "results", "seasons",
        "sprint_results", "status"
    ]
}
```

[141]:
```
# Visualize shape and sample of each dataset
for name, df in dataframes.items():
    print(f"\n==== {name.upper()} ====")
    print(f"Shape: {df.shape}")
    print(df.head())
```

```
==== CIRCUITS ====
Shape: (77, 9)
   circuitId   circuitRef                            name      location  \
0          1  albert_park  Albert Park Grand Prix Circuit     Melbourne
1          2       sepang    Sepang International Circuit  Kuala Lumpur
2          3      bahrain   Bahrain International Circuit        Sakhir
3          4    catalunya  Circuit de Barcelona-Catalunya      Montmeló
4          5     istanbul                    Istanbul Park      Istanbul

     country       lat        lng  alt  \
0  Australia -37.84970  144.96800   10
1   Malaysia   2.76083  101.73800   18
2    Bahrain  26.03250   50.51060    7
3      Spain  41.57000    2.26111  109
4     Turkey  40.95170   29.40500  130

                                                 url
0  http://en.wikipedia.org/wiki/Melbourne_Grand_P…
1  http://en.wikipedia.org/wiki/Sepang_Internatio…
2  http://en.wikipedia.org/wiki/Bahrain_Internati…
3  http://en.wikipedia.org/wiki/Circuit_de_Barcel…
4         http://en.wikipedia.org/wiki/Istanbul_Park

==== CONSTRUCTOR_RESULTS ====
Shape: (12625, 5)
```

```
   constructorResultsId  raceId  constructorId  points status
0                     1      18              1    14.0     \N
1                     2      18              2     8.0     \N
2                     3      18              3     9.0     \N
3                     4      18              4     5.0     \N
4                     5      18              5     2.0     \N

==== CONSTRUCTOR_STANDINGS ====
Shape: (13391, 7)
   constructorStandingsId  raceId  constructorId  points  position  \
0                       1      18              1    14.0         1
1                       2      18              2     8.0         3
2                       3      18              3     9.0         2
3                       4      18              4     5.0         4
4                       5      18              5     2.0         5

  positionText  wins
0            1     1
1            3     0
2            2     0
3            4     0
4            5     0

==== CONSTRUCTORS ====
Shape: (212, 5)
   constructorId constructorRef        name nationality  \
0              1        mclaren     McLaren     British
1              2     bmw_sauber  BMW Sauber      German
2              3       williams    Williams     British
3              4        renault     Renault      French
4              5     toro_rosso  Toro Rosso     Italian

                                                url
0              http://en.wikipedia.org/wiki/McLaren
1           http://en.wikipedia.org/wiki/BMW_Sauber
2  http://en.wikipedia.org/wiki/Williams_Grand_Pr…
3  http://en.wikipedia.org/wiki/Renault_in_Formul…
4   http://en.wikipedia.org/wiki/Scuderia_Toro_Rosso

==== DRIVER_STANDINGS ====
Shape: (34863, 7)
   driverStandingsId  raceId  driverId  points  position positionText  wins
0                  1      18         1    10.0         1            1     1
1                  2      18         2     8.0         2            2     0
2                  3      18         3     6.0         3            3     0
3                  4      18         4     5.0         4            4     0
4                  5      18         5     4.0         5            5     0
```

```
==== DRIVERS ====
Shape: (861, 9)
   driverId   driverRef number code  forename     surname         dob  \
0         1    hamilton     44  HAM     Lewis    Hamilton  1985-01-07
1         2    heidfeld     \N  HEI      Nick    Heidfeld  1977-05-10
2         3     rosberg      6  ROS      Nico     Rosberg  1985-06-27
3         4      alonso     14  ALO  Fernando      Alonso  1981-07-29
4         5  kovalainen     \N  KOV    Heikki  Kovalainen  1981-10-19

  nationality                                            url
0     British     http://en.wikipedia.org/wiki/Lewis_Hamilton
1      German      http://en.wikipedia.org/wiki/Nick_Heidfeld
2      German       http://en.wikipedia.org/wiki/Nico_Rosberg
3     Spanish    http://en.wikipedia.org/wiki/Fernando_Alonso
4     Finnish  http://en.wikipedia.org/wiki/Heikki_Kovalainen

==== LAP_TIMES ====
Shape: (589081, 6)
   raceId  driverId  lap  position      time  milliseconds
0     841        20    1         1  1:38.109         98109
1     841        20    2         1  1:33.006         93006
2     841        20    3         1  1:32.713         92713
3     841        20    4         1  1:32.803         92803
4     841        20    5         1  1:32.342         92342

==== PIT_STOPS ====
Shape: (11371, 7)
   raceId  driverId  stop  lap      time duration  milliseconds
0     841       153     1    1  17:05:23   26.898         26898
1     841        30     1    1  17:05:52   25.021         25021
2     841        17     1   11  17:20:48   23.426         23426
3     841         4     1   12  17:22:34   23.251         23251
4     841        13     1   13  17:24:10   23.842         23842

==== QUALIFYING ====
Shape: (10494, 9)
   qualifyId  raceId  driverId  constructorId  number  position        q1  \
0          1      18         1              1      22         1  1:26.572
1          2      18         9              2       4         2  1:26.103
2          3      18         5              1      23         3  1:25.664
3          4      18        13              6       2         4  1:25.994
4          5      18         2              2       3         5  1:25.960

         q2        q3
0  1:25.187  1:26.714
1  1:25.315  1:26.869
2  1:25.452  1:27.079
3  1:25.691  1:27.178
```

```
4  1:25.518  1:27.236

==== RACES ====
Shape: (1125, 18)
   raceId  year  round  circuitId                   name        date  \
0       1  2009      1          1  Australian Grand Prix  2009-03-29
1       2  2009      2          2   Malaysian Grand Prix  2009-04-05
2       3  2009      3         17     Chinese Grand Prix  2009-04-19
3       4  2009      4          3     Bahrain Grand Prix  2009-04-26
4       5  2009      5          4     Spanish Grand Prix  2009-05-10

       time                                                url fp1_date  \
0  06:00:00  http://en.wikipedia.org/wiki/2009_Australian_G...       \N
1  09:00:00  http://en.wikipedia.org/wiki/2009_Malaysian_Gr...       \N
2  07:00:00  http://en.wikipedia.org/wiki/2009_Chinese_Gran...       \N
3  12:00:00  http://en.wikipedia.org/wiki/2009_Bahrain_Gran...       \N
4  12:00:00  http://en.wikipedia.org/wiki/2009_Spanish_Gran...       \N

  fp1_time fp2_date fp2_time fp3_date fp3_time quali_date quali_time  \
0       \N       \N       \N       \N       \N         \N         \N
1       \N       \N       \N       \N       \N         \N         \N
2       \N       \N       \N       \N       \N         \N         \N
3       \N       \N       \N       \N       \N         \N         \N
4       \N       \N       \N       \N       \N         \N         \N

  sprint_date sprint_time
0          \N          \N
1          \N          \N
2          \N          \N
3          \N          \N
4          \N          \N

==== RESULTS ====
Shape: (26759, 18)
   resultId  raceId  driverId  constructorId number  grid position  \
0         1      18         1              1     22     1        1
1         2      18         2              2      3     5        2
2         3      18         3              3      7     7        3
3         4      18         4              4      5    11        4
4         5      18         5              1     23     3        5

  positionText  positionOrder  points  laps         time milliseconds  \
0            1              1    10.0    58  1:34:50.616      5690616
1            2              2     8.0    58       +5.478      5696094
2            3              3     6.0    58       +8.163      5698779
3            4              4     5.0    58      +17.181      5707797
4            5              5     4.0    58      +18.014      5708630
```

```
  fastestLap rank fastestLapTime fastestLapSpeed  statusId
0         39    2       1:27.452         218.300         1
1         41    3       1:27.739         217.586         1
2         41    5       1:28.090         216.719         1
3         58    7       1:28.603         215.464         1
4         43    1       1:27.418         218.385         1

==== SEASONS ====
Shape: (75, 2)
   year                                                url
0  2009  http://en.wikipedia.org/wiki/2009_Formula_One_…
1  2008  http://en.wikipedia.org/wiki/2008_Formula_One_…
2  2007  http://en.wikipedia.org/wiki/2007_Formula_One_…
3  2006  http://en.wikipedia.org/wiki/2006_Formula_One_…
4  2005  http://en.wikipedia.org/wiki/2005_Formula_One_…

==== SPRINT_RESULTS ====
Shape: (360, 16)
   resultId  raceId  driverId  constructorId  number  grid position  \
0         1    1061       830              9      33     2        1
1         2    1061         1            131      44     1        2
2         3    1061       822            131      77     3        3
3         4    1061       844              6      16     4        4
4         5    1061       846              1       4     6        5

  positionText  positionOrder  points  laps       time milliseconds  \
0            1              1       3    17  25:38.426      1538426
1            2              2       2    17     +1.430      1539856
2            3              3       1    17     +7.502      1545928
3            4              4       0    17    +11.278      1549704
4            5              5       0    17    +24.111      1562537

  fastestLap fastestLapTime  statusId
0         14       1:30.013         1
1         17       1:29.937         1
2         17       1:29.958         1
3         16       1:30.163         1
4         16       1:30.566         1

==== STATUS ====
Shape: (139, 2)
   statusId        status
0         1      Finished
1         2  Disqualified
2         3      Accident
3         4     Collision
4         5        Engine
```

To characterize "race-day performance," we are merging below tables:
results.csv
qualifying.csv
pit_stops.csv
lap_times.csv: average lap time per race

[142]:
```
pd.set_option('display.max_columns', None)
results = dataframes["results"][["raceId", "driverId", "constructorId", "grid",␣
↪"positionOrder", "points", "laps", "time", "milliseconds", "fastestLap",␣
↪"rank", "fastestLapTime", "fastestLapSpeed", "statusId"]].
↪rename(columns={"rank": "fastesLapRank", "milliseconds": "Total_time_ms",␣
↪"positionOrder":"finalPosition"}).merge(dataframes["races"][["raceId",␣
↪"year", "round", "circuitId", "name", "date"]], on="raceId", how="left")

#Merging with status table on statusId
results = results.merge(dataframes["status"], on="statusId", how="left")

# Merge with qualifying data
merged = results.merge(dataframes["qualifying"][["raceId", "driverId",␣
↪"position", "q1", "q2", "q3"]].rename(columns={"position":␣
↪"qualifyingPosition"}), on=["raceId", "driverId"], how="left")

merged = merged.merge(dataframes["drivers"][["driverId", "driverRef"]],␣
↪on="driverId", how="left")

#Merge with pit_stops data
#df_pit = dataframes["pit_stops"].rename(columns={"stop": "pit_stop_number",␣
↪"lap": "pit_stop_lap", "time":"pit_stop_time", "duration":␣
↪"pit_stop_duration","milliseconds": "pit_stop_duration_ms"})
#merged = merged.merge(df_pit, on=["raceId", "driverId"], how="left")

pit_summary = dataframes["pit_stops"].copy()
pit_summary["milliseconds"] = pd.to_numeric(pit_summary["milliseconds"],␣
↪errors="coerce")
pit_summary = pit_summary.groupby(["raceId", "driverId"]).agg(
    total_pit_stops=('stop', 'count'),
    average_pit_duration_ms=('milliseconds', 'mean')
).reset_index()
merged = merged.merge(pit_summary, on=["raceId", "driverId"], how="left")

#Merge with lap_times data
avg_lap_time = dataframes["lap_times"].groupby(["raceId",␣
↪"driverId"])["milliseconds"].mean().reset_index(name="avg_lap_time")
avg_lap_time["avg_lap_time_str"] = avg_lap_time["avg_lap_time"].apply(
    lambda ms: f"{int(ms // 60000)}:{(ms % 60000) / 1000:.3f}" if pd.notna(ms)␣
↪else None
)
```

```
merged = merged.merge(avg_lap_time.rename(columns = {"avg_lap_time":␣
↪"avg_lap_time_ms"}), on=["raceId", "driverId"], how="left")
```

[143]:
```
merged.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 26759 entries, 0 to 26758
Data columns (total 29 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   raceId                    26759 non-null  int64
 1   driverId                  26759 non-null  int64
 2   constructorId             26759 non-null  int64
 3   grid                      26759 non-null  int64
 4   finalPosition             26759 non-null  int64
 5   points                    26759 non-null  float64
 6   laps                      26759 non-null  int64
 7   time                      26759 non-null  object
 8   Total_time_ms             26759 non-null  object
 9   fastestLap                26759 non-null  object
 10  fastesLapRank             26759 non-null  object
 11  fastestLapTime            26759 non-null  object
 12  fastestLapSpeed           26759 non-null  object
 13  statusId                  26759 non-null  int64
 14  year                      26759 non-null  int64
 15  round                     26759 non-null  int64
 16  circuitId                 26759 non-null  int64
 17  name                      26759 non-null  object
 18  date                      26759 non-null  object
 19  status                    26759 non-null  object
 20  qualifyingPosition        10494 non-null  float64
 21  q1                        10494 non-null  object
 22  q2                        10472 non-null  object
 23  q3                        10448 non-null  object
 24  driverRef                 26759 non-null  object
 25  total_pit_stops           5575 non-null   float64
 26  average_pit_duration_ms   5575 non-null   float64
 27  avg_lap_time_ms           11041 non-null  float64
 28  avg_lap_time_str          11041 non-null  object
dtypes: float64(5), int64(10), object(14)
memory usage: 5.9+ MB
```

[144]:
```
merged.isnull().sum()
```

[144]:
```
raceId                     0
driverId                   0
constructorId              0
```

```
grid                            0
finalPosition                   0
points                          0
laps                            0
time                            0
Total_time_ms                   0
fastestLap                      0
fastesLapRank                   0
fastestLapTime                  0
fastestLapSpeed                 0
statusId                        0
year                            0
round                           0
circuitId                       0
name                            0
date                            0
status                          0
qualifyingPosition          16265
q1                          16265
q2                          16287
q3                          16311
driverRef                       0
total_pit_stops             21184
average_pit_duration_ms     21184
avg_lap_time_ms             15718
avg_lap_time_str            15718
dtype: int64
```

1.1.2 Data Cleaning and Transformation

The columns contains nearly a 50% of missing data. These columns is essential for clustering drivers based on performance. Imputing such a large percentage of missing values might introduce significant bias or noise into the analysis.

[145]:
```
#dropping the NaN values
merged = merged.dropna(subset=['qualifyingPosition', 'q1', 'q2', 'q3',␣
  ↪'total_pit_stops', 'average_pit_duration_ms', 'avg_lap_time_ms',␣
  ↪'avg_lap_time_str']).reset_index(drop=True)
```

[146]:
```
merged.isnull().sum().sum()
```

[146]: 0

[147]:
```
#Removing the values with \N in the data.
merged.replace(to_replace='\\N', value=np.nan, inplace=True)
merged.dropna(inplace=True)
```

[148]:
```
merged.shape
```

[148]: (2020, 29)

[149]:
```
#Checking duplicates
merged[merged.duplicated()]
```

[149]: Empty DataFrame
Columns: [raceId, driverId, constructorId, grid, finalPosition, points, laps, time, Total_time_ms, fastestLap, fastesLapRank, fastestLapTime, fastestLapSpeed, statusId, year, round, circuitId, name, date, status, qualifyingPosition, q1, q2, q3, driverRef, total_pit_stops, average_pit_duration_ms, avg_lap_time_ms, avg_lap_time_str]
Index: []

[150]:
```
merged['year'].unique()
```

[150]:
```
array([2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021,
       2022, 2023, 2024], dtype=int64)
```

[151]:
```
# position change between starting grid position and the final position
merged["position_change"] = merged["grid"] - merged["finalPosition"].astype(int)
```

[152]:
```
merged.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 2020 entries, 0 to 5500
Data columns (total 30 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   raceId                   2020 non-null   int64
 1   driverId                 2020 non-null   int64
 2   constructorId            2020 non-null   int64
 3   grid                     2020 non-null   int64
 4   finalPosition            2020 non-null   int64
 5   points                   2020 non-null   float64
 6   laps                     2020 non-null   int64
 7   time                     2020 non-null   object
 8   Total_time_ms            2020 non-null   object
 9   fastestLap               2020 non-null   object
 10  fastesLapRank            2020 non-null   object
 11  fastestLapTime           2020 non-null   object
 12  fastestLapSpeed          2020 non-null   object
 13  statusId                 2020 non-null   int64
 14  year                     2020 non-null   int64
 15  round                    2020 non-null   int64
 16  circuitId                2020 non-null   int64
 17  name                     2020 non-null   object
 18  date                     2020 non-null   object
 19  status                   2020 non-null   object
```

```
 20  qualifyingPosition       2020 non-null   float64
 21  q1                       2020 non-null   object
 22  q2                       2020 non-null   object
 23  q3                       2020 non-null   object
 24  driverRef                2020 non-null   object
 25  total_pit_stops          2020 non-null   float64
 26  average_pit_duration_ms  2020 non-null   float64
 27  avg_lap_time_ms          2020 non-null   float64
 28  avg_lap_time_str         2020 non-null   object
 29  position_change          2020 non-null   int64
dtypes: float64(5), int64(11), object(14)
memory usage: 489.2+ KB
```

[153]:
```
#merged["pit_stop_time"] = pd.to_datetime(merged["pit_stop_time"],␣
↪errors="coerce", format="%H:%M:%S").dt.time
```

[154]:
```
merged.select_dtypes(include="object").columns
```

[154]:
```
Index(['time', 'Total_time_ms', 'fastestLap', 'fastesLapRank',
       'fastestLapTime', 'fastestLapSpeed', 'name', 'date', 'status', 'q1',
       'q2', 'q3', 'driverRef', 'avg_lap_time_str'],
      dtype='object')
```

[155]:
```
def time_str_to_ms(t):
    if pd.isna(t): return np.nan
    t = str(t).strip()
    if re.match(r"^\d+:\d+\.\d+$", t):  # Matches '1:29.844' style
        minutes, rest = t.split(":")
        seconds = float(rest)
        return int(minutes) * 60000 + seconds * 1000

# Apply to time columns of type object to convert into milliseconds of type␣
↪float.
merged["fastestLapTime_ms"] = merged["fastestLapTime"].apply(time_str_to_ms).
↪astype("Int64")
merged["q1_ms"] = merged["q1"].apply(time_str_to_ms).astype("Int64")
merged["q2_ms"] = merged["q2"].apply(time_str_to_ms).astype("Int64")
merged["q3_ms"] = merged["q3"].apply(time_str_to_ms).astype("Int64")
merged["Total_time_ms"] = pd.to_numeric(merged["Total_time_ms"],␣
↪errors="coerce").astype("Int64")
```

[156]:
```
merged["fastestLap"] = pd.to_numeric(merged["fastestLap"], errors="coerce").
↪astype("Int64")
merged["fastesLapRank"] = pd.to_numeric(merged["fastesLapRank"],␣
↪errors="coerce").astype("Int64")
merged["fastestLapSpeed"] = pd.to_numeric(merged["fastestLapSpeed"],␣
↪errors="coerce").astype("float64")
```

[157]:
```
#Encoding the qualitative variable status
le = LabelEncoder()
merged["status_encoded"] = le.fit_transform(merged["status"])
print(dict(zip(le.classes_, le.transform(le.classes_))))
```

```
{'Disqualified': 0, 'Finished': 1, 'Power Unit': 2}
```

[158]:
```
final_data = merged.drop(columns = ['time', 'fastestLapTime', 'q1', 'q2', 'q3',␣
↪'driverRef', 'avg_lap_time_str', 'name', 'status', 'date'])
```

Columns like raceId, driverId, and constructorId are numerically stored, but: They are still categorical identifiers — just with numeric format. Their values do not carry mathematical meaning (e.g., driverId = 10 isn't "twice" driverId = 5). So removing those columns and perfroming SVD on the remaining features.

[159]:
```
scaler = StandardScaler()
X_features = final_data.drop(columns=["raceId", "driverId", "constructorId",␣
↪"circuitId", "statusId", "round", "status_encoded", "year", "laps"])
X_scaled = StandardScaler().fit_transform(X_features)
data_scaled_df = pd.DataFrame(X_scaled, columns=X_features.columns)
```

[160]:
```
final_data.head()
```

[160]:
```
   raceId  driverId  constructorId  grid  finalPosition  points  laps  \
0     841        20              9     1              1    25.0    58
1     841         1              1     2              2    18.0    58
2     841       808              4     6              3    15.0    58
3     841         4              6     5              4    12.0    58
4     841        17              9     3              5    10.0    58

   Total_time_ms  fastestLap  fastesLapRank  fastestLapSpeed  statusId  year  \
0        5370259          44              4          212.488         1  2011
1        5392556          41              8          211.382         1  2011
2        5400819          55              7          211.969         1  2011
3        5402031          49              2          213.336         1  2011
4        5408430          50              3          213.066         1  2011

   round  circuitId  qualifyingPosition  total_pit_stops  \
0      1          1                 1.0              2.0
1      1          1                 2.0              2.0
2      1          1                 6.0              2.0
3      1          1                 5.0              3.0
4      1          1                 3.0              3.0

   average_pit_duration_ms  avg_lap_time_ms  position_change  \
0             23319.500000     92590.672414                0
1             23213.000000     92975.103448                0
2             25109.000000     93117.568966                3
```

```
3            24055.000000     93138.465517                 1
4            24058.666667     93248.793103                -2

   fastestLapTime_ms  q1_ms  q2_ms  q3_ms  status_encoded
0              89844  85296  84090  83529               1
1              90314  85384  84595  84307               1
2              90064  85543  85582  85247               1
3              89487  85707  85242  84974               1
4              89600  85900  84658  84395               1
```

[161]:
```python
data_scaled_df.head()
```

[161]:
```
       grid  finalPosition    points  Total_time_ms  fastestLap  \
0 -1.290033      -1.235402  1.737068      -0.439229   -0.391093
1 -0.964747      -0.912358  0.794654      -0.420336   -0.635854
2  0.336397      -0.589315  0.390762      -0.413335    0.506365
3  0.011111      -0.266271 -0.013130      -0.412308    0.016842
4 -0.639461       0.056772 -0.282391      -0.406886    0.098430

   fastesLapRank  fastestLapSpeed  qualifyingPosition  total_pit_stops  \
0      -0.513603         0.236956           -1.414751        -0.027390
1       0.465612         0.184532           -1.046952        -0.027390
2       0.220808         0.212356            0.424243        -0.027390
3      -1.003211         0.277152            0.056444         1.016529
4      -0.758407         0.264354           -0.679154         1.016529

   average_pit_duration_ms  avg_lap_time_ms  position_change  \
0                -0.269378        -0.389816        -0.048314
1                -0.269930        -0.369424        -0.048314
2                -0.260102        -0.361867         0.975400
3                -0.265566        -0.360759         0.292924
4                -0.265547        -0.354907        -0.730789

   fastestLapTime_ms     q1_ms     q2_ms     q3_ms
0          -0.065757 -0.250179 -0.293561 -0.325067
1          -0.025557 -0.243077 -0.252019 -0.262457
2          -0.046940 -0.230246 -0.170826 -0.186809
3          -0.096292 -0.217011 -0.198795 -0.208779
4          -0.086627 -0.201436 -0.246836 -0.255375
```

2 MATRIX COMPLETION

[162]:
```python
# Created a boolean mask 'non_nan' that is True for each entry in X that isn't␣
↪NaN (i.e., observed data) and False for missing values

def matrix_completion_svd(
    X: np.ndarray,
```

```python
    rank: int = 5,
    max_iters: int = 100,
    tol: float = 1e-6,
    verbose: bool = False
) -> np.ndarray:
    non_nan = ~np.isnan(X)
    X_filled = X.copy()

    # Finding all missing entries in X_filled and replaces each with the mean␣
↪of its respective column.
    col_means = np.nanmean(X, axis=0)
    missing_idx = np.where(~non_nan)
    X_filled[missing_idx] = np.take(col_means, missing_idx[1])

    for it in range(1, max_iters + 1):
        # Constructing low-rank matrix again
        U, s, Vt = np.linalg.svd(X_filled, full_matrices=False)
        S = np.diag(s[:rank])
        X_lowrank = U[:, :rank] @ S @ Vt[:rank, :]
        X_new = X_lowrank.copy()
        X_new[non_nan] = X[non_nan]

        # Checking convergence
        rel_change = np.linalg.norm(X_new - X_filled) / np.linalg.norm(X_filled)
        if verbose:
            print(f"[Iter {it:3d}/{max_iters}] rel-change = {rel_change:.2e}")

        X_filled = X_new
        if rel_change < tol:
            if verbose:
                print("Converged.")
            break

    return X_filled


# Pivot the results DataFrame into a driver-by-race points matrix and then␣
↪converting it to a NumPy array with NaNs for missing scores
results_df = dataframes['results']
matrix_df = (
    results_df
    .pivot_table(
        index='driverId',
        columns='raceId',
        values='points'
    )
)
```

```python
X = matrix_df.values

X_completed = matrix_completion_svd(X, rank=5, max_iters=100, tol=1e-6,␣
 ↪verbose=True)

completed_df = pd.DataFrame(
    X_completed,
    index=matrix_df.index,
    columns=matrix_df.columns
)

print("Missing before:", np.isnan(X).sum())
print("Missing after: ", np.isnan(X_completed).sum())
```

```
[Iter   1/100] rel-change = 5.06e-02
[Iter   2/100] rel-change = 3.07e-02
[Iter   3/100] rel-change = 2.32e-02
[Iter   4/100] rel-change = 1.92e-02
[Iter   5/100] rel-change = 1.66e-02
[Iter   6/100] rel-change = 1.45e-02
[Iter   7/100] rel-change = 1.28e-02
[Iter   8/100] rel-change = 1.15e-02
[Iter   9/100] rel-change = 1.05e-02
[Iter  10/100] rel-change = 9.84e-03
[Iter  11/100] rel-change = 9.39e-03
[Iter  12/100] rel-change = 9.08e-03
[Iter  13/100] rel-change = 8.89e-03
[Iter  14/100] rel-change = 8.78e-03
[Iter  15/100] rel-change = 8.74e-03
[Iter  16/100] rel-change = 8.74e-03
[Iter  17/100] rel-change = 8.76e-03
[Iter  18/100] rel-change = 8.75e-03
[Iter  19/100] rel-change = 8.68e-03
[Iter  20/100] rel-change = 8.54e-03
[Iter  21/100] rel-change = 8.32e-03
[Iter  22/100] rel-change = 8.07e-03
[Iter  23/100] rel-change = 7.81e-03
[Iter  24/100] rel-change = 7.60e-03
[Iter  25/100] rel-change = 7.43e-03
[Iter  26/100] rel-change = 7.33e-03
[Iter  27/100] rel-change = 7.26e-03
[Iter  28/100] rel-change = 7.23e-03
[Iter  29/100] rel-change = 7.21e-03
[Iter  30/100] rel-change = 7.18e-03
[Iter  31/100] rel-change = 7.12e-03
[Iter  32/100] rel-change = 7.04e-03
[Iter  33/100] rel-change = 6.92e-03
[Iter  34/100] rel-change = 6.77e-03
```

```
[Iter  35/100] rel-change = 6.61e-03
[Iter  36/100] rel-change = 6.44e-03
[Iter  37/100] rel-change = 6.27e-03
[Iter  38/100] rel-change = 6.11e-03
[Iter  39/100] rel-change = 5.96e-03
[Iter  40/100] rel-change = 5.83e-03
[Iter  41/100] rel-change = 5.70e-03
[Iter  42/100] rel-change = 5.58e-03
[Iter  43/100] rel-change = 5.46e-03
[Iter  44/100] rel-change = 5.35e-03
[Iter  45/100] rel-change = 5.24e-03
[Iter  46/100] rel-change = 5.14e-03
[Iter  47/100] rel-change = 5.05e-03
[Iter  48/100] rel-change = 4.95e-03
[Iter  49/100] rel-change = 4.87e-03
[Iter  50/100] rel-change = 4.79e-03
[Iter  51/100] rel-change = 4.72e-03
[Iter  52/100] rel-change = 4.65e-03
[Iter  53/100] rel-change = 4.58e-03
[Iter  54/100] rel-change = 4.52e-03
[Iter  55/100] rel-change = 4.47e-03
[Iter  56/100] rel-change = 4.41e-03
[Iter  57/100] rel-change = 4.35e-03
[Iter  58/100] rel-change = 4.29e-03
[Iter  59/100] rel-change = 4.23e-03
[Iter  60/100] rel-change = 4.17e-03
[Iter  61/100] rel-change = 4.11e-03
[Iter  62/100] rel-change = 4.05e-03
[Iter  63/100] rel-change = 3.99e-03
[Iter  64/100] rel-change = 3.93e-03
[Iter  65/100] rel-change = 3.88e-03
[Iter  66/100] rel-change = 3.83e-03
[Iter  67/100] rel-change = 3.78e-03
[Iter  68/100] rel-change = 3.74e-03
[Iter  69/100] rel-change = 3.70e-03
[Iter  70/100] rel-change = 3.67e-03
[Iter  71/100] rel-change = 3.64e-03
[Iter  72/100] rel-change = 3.61e-03
[Iter  73/100] rel-change = 3.58e-03
[Iter  74/100] rel-change = 3.56e-03
[Iter  75/100] rel-change = 3.54e-03
[Iter  76/100] rel-change = 3.51e-03
[Iter  77/100] rel-change = 3.49e-03
[Iter  78/100] rel-change = 3.47e-03
[Iter  79/100] rel-change = 3.45e-03
[Iter  80/100] rel-change = 3.43e-03
[Iter  81/100] rel-change = 3.41e-03
[Iter  82/100] rel-change = 3.39e-03
```

```
[Iter  83/100] rel-change = 3.37e-03
[Iter  84/100] rel-change = 3.35e-03
[Iter  85/100] rel-change = 3.33e-03
[Iter  86/100] rel-change = 3.31e-03
[Iter  87/100] rel-change = 3.29e-03
[Iter  88/100] rel-change = 3.28e-03
[Iter  89/100] rel-change = 3.26e-03
[Iter  90/100] rel-change = 3.25e-03
[Iter  91/100] rel-change = 3.23e-03
[Iter  92/100] rel-change = 3.21e-03
[Iter  93/100] rel-change = 3.20e-03
[Iter  94/100] rel-change = 3.18e-03
[Iter  95/100] rel-change = 3.17e-03
[Iter  96/100] rel-change = 3.15e-03
[Iter  97/100] rel-change = 3.13e-03
[Iter  98/100] rel-change = 3.12e-03
[Iter  99/100] rel-change = 3.10e-03
[Iter 100/100] rel-change = 3.08e-03
Missing before: 941957
Missing after:  0
```

[163]:
```python
completed_df.sample(5)
```

[163]:
```
raceId          1         2         3         4         5         6   \
driverId
292       1.814739  0.786134  1.048315  1.775408  1.526402  1.981822
475       4.575409  1.586142  0.366874  2.478301  3.401873  5.159508
592       1.853458  0.791756  1.018651  1.763092  1.563269  2.036441
94        1.070475  0.543267  1.059953  1.560002  1.104981  1.166037
771       1.936208  0.819481  1.010812  1.792873  1.613124  2.125509

raceId          7         8         9         10        11        12  \
driverId
292       0.972037  1.129973  1.026457  1.924568  2.648174  1.659775
475       1.743818  0.839924  1.220544  2.336195  4.303195  3.499378
592       0.992878  1.108163  1.028112  1.888506  2.622518  1.680069
94        0.754489  0.974267  0.833684  1.415934  1.856763  1.192180
771       1.013251  1.111511  1.039051  1.932180  2.703608  1.730684

raceId          13        14        15        16        17        18  \
driverId
292       2.121762  2.175089  1.824604  1.260683  1.007089  2.020801
475       5.427427  2.719848  1.192185  1.260261  0.848841  2.333121
592       2.172447  2.147189  1.761493  1.251446  0.995033  1.974924
94        1.335904  1.957102  1.651306  1.183665  0.959248  1.613380
771       2.264141  2.186021  1.772075  1.261654  0.994754  2.019753
```

```
raceId         19        20        21        22        23        24  \
driverId
292       1.745402  1.687956  1.871888  2.131866  1.538772  1.355564
475       3.747766  4.047992  3.725576  3.595823  0.618983  2.350418
592       1.765451  1.722964  1.892485  2.131649  1.482830  1.352668
94        1.215117  1.144969  1.427547  1.738480  1.563324  1.024512
771       1.827312  1.789696  1.950947  2.189384  1.486792  1.392419

raceId         25        26        27        28        29        30  \
driverId
292       1.639756  2.394658  2.020288  1.775563  1.737482  1.829892
475       4.168735  3.253206  2.003001  3.872328  1.568523  2.495527
592       1.682554  2.351833  1.966459  1.793981  1.692671  1.817212
94        1.077378  1.900501  1.804426  1.209195  1.625872  1.696334
771       1.751278  2.409891  1.999344  1.861157  1.715941  1.850262

raceId         31        32        33        34        35        36  \
driverId
292       1.283276  1.838163  1.658804  2.344105  2.017779  2.309052
475       1.481532  2.848424  4.924094  3.870784  4.182831  3.530001
592       1.275173  1.829827  1.736429  2.338178  2.049397  2.310880
94        1.299202  1.397540  1.136393  1.942663  1.639933  2.083917
771       1.283696  1.877542  1.810315  2.400432  2.108911  2.346703

raceId         37        38        39        40        41        42  \
driverId
292       2.284687  2.203220  1.894653  2.012765  1.975772  2.144922
475       3.673447  2.481790  3.486211  4.213475  0.357582  3.780114
592       2.289686  2.175099  1.899214  2.043158  1.887827  2.153528
94        2.057723  2.161225  1.421705  1.625677  2.163500  1.770570
771       2.333764  2.196838  1.964996  2.111087  1.872602  2.212401

raceId         43        44        45        46        47        48  \
driverId
292       2.170760  2.245304  1.304335  2.228423  2.141819  2.190798
475       2.332903  3.547132  3.739811  3.715617  4.831649  4.590076
592       2.146490  2.254451  1.367500  2.220172  2.177575  2.218464
94        2.203145  2.078422  0.902177  1.746572  1.546514  1.625557
771       2.156390  2.289153  1.425912  2.281846  2.256904  2.294185

raceId         49        50        51        52        53        54  \
driverId
292       2.135568  1.897079  1.876986  2.079719  1.054363  1.408579
475       4.756592  2.301497  5.238711  5.261249  5.787261  4.742641
592       2.170576  1.858787  1.952534  2.132927  1.188259  1.489427
94        1.505365  1.561873  1.344539  1.350282  0.236170  0.901413
771       2.248735  1.898771  2.027515  2.221007  1.287918  1.569188
```

```
raceId          55        56        57        58        59        60    \
driverId
292       1.790871  1.118357  1.348269  1.255670  1.457083  1.323940
475       4.215889  5.776382  5.620638  5.834662  4.756677  5.950021
592       1.838696  1.241578  1.441718  1.371292  1.524051  1.440766
94        1.393165  0.277134  0.417902  0.447581  0.759200  0.518936
771       1.892067  1.348563  1.551231  1.476195  1.611522  1.545309

raceId          61        62        63        64        65        66    \
driverId
292       1.224862  1.210725  1.417270  1.499524  2.117559  2.077444
475       5.673236  4.496808  6.388352  6.266826  2.462214  5.221316
592       1.337375  1.269433  1.534229  1.611113  2.079274  2.128691
94        0.431317  0.404524  0.459287  0.593876  1.866757  1.460698
771       1.436953  1.366188  1.655697  1.726719  2.115925  2.223445

raceId          67        68        69        70        71        72    \
driverId
292       1.019059  1.183322  2.203045  2.081111  1.579986  1.541099
475       5.638693  5.998535  4.934609  5.670763  2.747357  3.998041
592       1.132426  1.299847  2.254062  2.151857  1.582825  1.582984
94        0.085020  0.243539  1.815423  1.444826  1.098404  1.024403
771       1.238160  1.418327  2.323771  2.246828  1.627646  1.650847

raceId          73        74        75        76        77        78    \
driverId
292       1.504336  1.608580  1.615538  1.824155  1.696132  1.735895
475       4.114039  4.604705  4.777802  4.107357  4.097749  4.032415
592       1.567572  1.662054  1.700495  1.874038  1.725029  1.757513
94        1.084477  0.893961  1.167629  1.372901  1.013830  0.833424
771       1.622557  1.752381  1.757821  1.922854  1.801523  1.834015

raceId          79        80        81        82        83        84    \
driverId
292       1.507336  1.277060  1.414468  1.298959  1.431448  1.458562
475       3.250402  5.897297  5.877035  5.481992  4.543640  5.124866
592       1.485234  1.401372  1.530609  1.407540  1.490263  1.560191
94        0.403275  0.530472  0.718737  0.638015  0.594348  1.008485
771       1.580102  1.496286  1.625487  1.502758  1.570281  1.625806

raceId          85        86        87        88        89        90    \
driverId
292       1.507233  1.778627  1.537127  1.353656  1.548848  1.759097
475       4.945777  3.873833  6.058150  3.993115  4.304342  5.531743
592       1.591803  1.828861  1.644067  1.428894  1.620768  1.808390
94        1.028594  1.447023  0.683829  0.919151  1.208958  0.542103
```

```
771        1.664533  1.869013  1.747515  1.475727  1.671429  1.934025

raceId          91        92        93        94        95        96    \
driverId
292        1.760500  1.677404  1.473241  1.675189  2.036818  1.552739
475        5.158302  4.555866  5.881545  4.756939  2.724474  4.898006
592        1.801209  1.699954  1.562439  1.699538  2.003364  1.598728
94         0.644021  0.557573  0.408992  0.495497  1.655896  0.444677
771        1.918831  1.808300  1.684712  1.814115  2.052894  1.710120

raceId          97        98        99        100       101       102   \
driverId
292        1.588257  1.903862  1.611202  1.621167  1.464254  1.764394
475        4.759436  3.645349  5.746352  5.696552  4.795812  5.597675
592        1.624646  1.875725  1.683090  1.691665  1.529814  1.817672
94         0.451985  0.775986  0.432901  0.405439  0.550192  0.582197
771        1.730859  1.970734  1.805311  1.807713  1.628803  1.943310

raceId          103       104       105       106       107       108   \
driverId
292        1.849027  2.028106  1.726826  1.339712  1.786590  1.738333
475        4.829055  4.121853  4.595631  5.285462  5.064830  3.883732
592        1.879734  2.014044  1.778411  1.419065  1.844120  1.755725
94         0.819501  0.937886  1.108356  0.347903  1.020873  0.938819
771        1.974775  2.115153  1.850941  1.526894  1.929982  1.829502

raceId          109       110       111       112       113       114   \
driverId
292        1.742340  1.535890  1.787728  1.583109  1.705041  1.604964
475        4.973109  3.419152  5.410716  5.375434  4.669059  5.592135
592        1.797622  1.573280  1.836078  1.644935  1.738467  1.685335
94         0.924747  1.210718  0.568291  0.460856  0.574929  0.672475
771        1.879952  1.613659  1.949044  1.764275  1.836233  1.787503

raceId          115       116       117       118       119       120   \
driverId
292        1.597465  1.666409  1.728971  2.006835  1.655764  1.387862
475        5.487445  4.387690  3.916519  4.297774  3.310377  4.604792
592        1.662047  1.702344  1.743490  2.007245  1.664827  1.474479
94         0.439447  0.829746  0.827518  0.994189  1.109441  0.861050
771        1.780414  1.791264  1.823605  2.098224  1.727175  1.538180

raceId          121       122       123       124       125       126   \
driverId
292        1.947215  1.567737  1.879325  0.920871  1.024913  1.005883
475        5.048877  4.655456  2.942430  4.861007  3.895399  4.001204
592        1.974334  1.610563  1.857193  1.016209  1.085552  1.057306
```

94	0.805674	0.617683	1.207743	0.017389	0.416516	0.082919
771	2.086038	1.704222	1.907545	1.115194	1.161088	1.150557

raceId	127	128	129	130	131	132	\
driverId							
292	1.182130	1.109700	1.173980	1.102971	1.156578	1.322503	
475	4.178480	4.064586	4.126174	1.575744	3.327526	3.424443	
592	1.219393	1.157045	1.208487	1.070783	1.170516	1.333256	
94	0.122950	0.209697	0.105230	0.454005	0.174418	0.450806	
771	1.322867	1.252198	1.312319	1.117763	1.254664	1.411519	

raceId	133	134	135	136	137	138	\
driverId							
292	1.347712	1.035294	1.154978	1.391428	1.248493	1.323971	
475	3.622418	4.788336	4.296822	2.770121	3.737777	2.458248	
592	1.357412	1.116484	1.200345	1.372129	1.265778	1.300487	
94	0.392328	0.070167	0.125351	0.513984	0.197441	0.543998	
771	1.451293	1.214530	1.304304	1.443168	1.363490	1.367405	

raceId	139	140	141	142	143	144	\
driverId							
292	1.455007	1.180425	1.222635	1.237423	1.174982	1.224906	
475	2.284014	4.562204	2.270398	2.336120	0.780751	0.326275	
592	1.412867	1.234553	1.201899	1.216229	1.122020	1.161667	
94	0.595581	0.124365	0.317082	0.282917	0.619415	0.866912	
771	1.481678	1.338325	1.269166	1.287547	1.151664	1.172242	

raceId	145	146	147	148	149	150	\
driverId							
292	1.164070	1.247734	1.352853	0.935545	1.074264	1.151623	
475	3.039502	1.218994	2.376155	2.345832	3.044731	2.858275	
592	1.173776	1.203034	1.326566	0.950192	1.089183	1.147879	
94	0.362030	0.548170	0.487615	0.326218	0.129581	0.177071	
771	1.253249	1.240640	1.397558	1.000128	1.170815	1.228764	

raceId	151	152	153	154	155	156	\
driverId							
292	0.929992	1.379421	1.207656	0.977106	1.207948	0.836882	
475	1.477221	1.695603	2.381840	1.931516	3.069720	0.065141	
592	0.929710	1.341315	1.190315	0.969939	1.216456	0.804449	
94	0.377787	1.030781	0.242629	0.137878	0.566716	0.432926	
771	0.961921	1.377534	1.261865	1.027868	1.287356	0.810484	

raceId	157	158	159	160	161	162	\
driverId							
292	1.007845	1.255208	0.909277	1.167452	1.186840	1.361356	
475	2.665569	3.615890	4.937428	2.097802	1.100154	1.171932	

```
592       1.020357  1.269387  1.000513  1.161899  1.169287  1.341826
94        0.113484  0.282944 -0.019801  0.354078  0.688675  0.915294
771       1.092878  1.365405  1.107534  1.220073  1.184402  1.348603

raceId         163       164       165       166       167       168  \
driverId
292       1.036916  1.344248  1.133992  1.424277  1.440429  1.511374
475       2.883215  0.071020  4.519799 -0.939647  0.346128  2.314209
592       1.065949  1.269448  1.195760  1.324450  1.399403  1.507680
94        0.134859  0.981636  0.092111  1.198255  1.249678  0.998434
771       1.134760  1.268131  1.296789  1.301513  1.386623  1.534865

raceId         169       170       171       172       173       174  \
driverId
292       1.242041  1.145961  0.972103  1.240184  1.156654  1.201539
475       2.171409  3.173715  3.569710  1.969822  3.071392  1.548409
592       1.255025  1.204894  1.028647  1.203073  1.187037  1.166691
94        0.590599  0.521659  0.146505  0.396330  0.275710  0.362900
771       1.290889  1.251950  1.106248  1.274626  1.257909  1.222738

raceId         175       176       177       178       179       180  \
driverId
292       1.086817  1.032262  1.084488  0.878494  1.168188  1.055548
475       1.240195  2.559974  3.391107  2.997276  1.945728  2.092807
592       1.074501  1.082346  1.108678  0.929374  1.195035  1.104674
94        0.940884  0.555062  0.074156  0.184971  0.718439  0.928817
771       1.088513  1.120420  1.195898  0.999245  1.215116  1.111856

raceId         181       182       183       184       185       186  \
driverId
292       1.249149  1.298957  1.263163  1.073636  1.442610  1.437195
475       1.552464  1.126167  1.735440  0.614914  2.782076  2.370119
592       1.240136  1.271173  1.265310  1.045968  1.476243  1.447283
94        0.909301  1.004333  0.822108  1.043932  0.782912  0.745767
771       1.259962  1.279978  1.294475  1.044784  1.518217  1.488458

raceId         187       188       189       190       191       192  \
driverId
292       1.242285  1.134707  1.045277  1.067389  1.290083  1.095184
475       1.050172  0.887934  2.154175  2.690081  1.008380  1.800508
592       1.198776  1.088955  1.065004  1.121730  1.290609  1.120517
94        0.898125  0.840717  0.570830  0.591792  1.181332  0.673797
771       1.219077  1.104759  1.108337  1.161406  1.282550  1.140106

raceId         193       194       195       196       197       198  \
driverId
292       0.959020  1.219168  1.197913  1.055069  0.802695  1.033351
```

```
475       2.988024  1.190919  1.652581  1.844945  2.974822  2.655373
592       1.006251  1.184223  1.207093  1.098024  0.837067  1.062458
94        0.229138  0.680603  0.736375  1.010877  0.021156  0.379215
771       1.074469  1.220397  1.227423  1.101954  0.917339  1.129490

raceId         199       200       201       202       203       204  \
driverId
292       0.882300  1.183309  1.344005  1.152409  1.159580  0.968459
475       3.129318  1.884175  2.400630  3.130955  4.721447  3.434340
592       0.939993  1.208309  1.380355  1.177503  1.251326  1.021144
94        0.160195  0.760453  1.031637  0.348025  0.756858  0.187384
771       1.009583  1.227861  1.401487  1.256770  1.317246  1.100114

raceId         205       206       207       208       209       210  \
driverId
292       1.075307  1.252338  0.970665  1.481977  1.478365  0.951173
475       2.389183  2.326615  1.295312  0.968890  0.615107  2.113111
592       1.120282  1.302435  0.961346  1.435211  1.426429  0.965550
94        0.559807  1.089941  0.378981  1.402724  1.566728  0.530815
771       1.154341  1.311570  0.991323  1.457239  1.434850  1.014056

raceId         211       212       213       214       215       216  \
driverId
292       0.967617  1.511487  0.925028  1.118164  1.603626  0.938703
475       2.230745  0.636868  3.121843  2.551293  0.681647  2.508323
592       0.956071  1.436684  0.954112  1.123373  1.531748  0.966441
94        0.145721  1.430366  0.159022  0.462418  1.602117  0.317551
771       1.029273  1.464122  1.037464  1.197131  1.549121  1.019484

raceId         217       218       219       220       221       222  \
driverId
292       1.466541  0.892157  1.253048  1.521731  1.546595  0.920986
475       2.856121  2.791597  0.564978  0.252682  0.577693  2.703226
592       1.472753  0.916758  1.198899  1.443970  1.475274  0.952836
94        1.240208  0.211104  0.996501  1.471794  1.599444  0.257564
771       1.529564  0.993937  1.210597  1.454660  1.491959  1.021453

raceId         223       224       225       226       227       228  \
driverId
292       1.204131  1.376979  1.160250  1.472343  1.390828  1.152555
475       0.826259  4.393384  5.736286  4.503194  2.110194  5.646053
592       1.201779  1.461102  1.287727  1.536569  1.367939  1.263703
94        1.125402  1.162282  0.550279  1.152751  0.943104  0.349686
771       1.193740  1.518912  1.377774  1.605453  1.425968  1.365972

raceId         229       230       231       232       233       234  \
driverId
```

```
292        1.133395  1.077866  1.426837  1.444041  1.508792  1.415947
475       -0.034907  2.627633  4.268817  4.536374  0.690229  5.140018
592        1.069188  1.087810  1.497304  1.522396  1.454622  1.499804
94         1.043404  0.434417  1.155706  1.162773  1.427348  0.875780
771        1.064526  1.162828  1.555471  1.585323  1.465353  1.586225

raceId          235       236       237       238       239       240  \
driverId
292        1.595696  1.153719  0.973112  1.489577  1.089165  0.957986
475        2.940144  3.423239  3.041379  3.519871  6.044559  2.560910
592        1.612622  1.195529  1.002527  1.518823  1.234150  0.977929
94         1.522258  0.491550  0.163532  1.184682  0.335231  0.000000
771        1.654549  1.277530  1.080058  1.587939  1.332435  1.045545

raceId          241       242       243       244       245       246  \
driverId
292        1.113651  1.079443  1.094318  1.041473  1.332580  0.984182
475        5.506055  5.084942  3.525582  4.797775  1.325781  4.578594
592        1.225570  1.195921  1.131532  1.124642  1.279515  1.057938
94         0.000000  0.000000  0.000000  0.000000  0.000000  0.000000
771        1.320159  1.263417  1.220647  1.230118  1.316871  1.160095

raceId          247       248       249       250       251       252  \
driverId
292        1.257424  0.993279  0.983052  1.026934  1.096721  0.919417
475        0.289527  2.124890  4.272615  4.606365  0.702221  2.890417
592        1.190503  0.986409  1.072901  1.102054  1.079730  0.951714
94         0.000000  0.000000  0.486145  0.179439  1.161675  0.138905
771        1.199421  1.053958  1.130061  1.205421  1.079625  1.012256

raceId          253       254       255       256       257       258  \
driverId
292        1.086301  1.110775  0.989053  1.083664  0.991866  1.100497
475        2.328131  4.330713  2.565573  4.181322  4.677814  2.473939
592        1.071396  1.190237  1.020715  1.165657  1.061627  1.089945
94         0.246758  0.143412  0.396085  0.827173  0.000000  0.000000
771        1.147381  1.277072  1.083636  1.220730  1.170857  1.167001

raceId          259       260       261       262       263       264  \
driverId
292        0.893754  0.955893  0.995051  0.984304  0.910882  0.978784
475        2.753531  2.534794  5.708493  4.922480  4.790133  5.044787
592        0.928711  0.962851  1.121438  1.072035  1.001417  1.096475
94         0.000000  0.000000  2.000000  0.000000  2.000000  0.000000
771        0.999259  1.039752  1.222694  1.179806  1.107176  1.161289

raceId          265       266       267       268       269       270  \
```

```
driverId
292       0.939964  0.862880  0.764148  1.238626  1.053752  0.851092
475       0.659794  4.772771  4.808606  6.896293  5.690191  5.007089
592       0.906135  0.959540  0.905621  1.388068  1.180127  0.968690
94        0.000000  0.000000  0.000000  0.000000  0.000000  0.000000
771       0.917624  1.060364  0.955760  1.512258  1.272577  1.067841

raceId         271       272       273       274       275       276  \
driverId
292       0.877405  0.740111  0.764818  0.612179  0.612051  0.654614
475       5.691409  0.514119  1.556572  3.363338  3.454241  1.641124
592       1.002737  0.700209  0.801423  0.699102  0.707395  0.668865
94        0.000000  0.000000  0.574779  0.243860  0.288738  0.219961
771       1.106748  0.711513  0.819293  0.743164  0.751263  0.712858

raceId         277       278       279       280       281       282  \
driverId
292       0.596100  0.638945  0.600056  0.530887  0.639453  0.639014
475       1.965584  3.640879  3.125748  3.850229  1.867235  1.881451
592       0.634722  0.738878  0.658974  0.625689  0.659473  0.667885
94        0.107123  0.342029  0.057199  0.036778  0.000000  0.000000
771       0.674079  0.784469  0.729477  0.699454  0.707436  0.711312

raceId         283       284       285       286       287       288  \
driverId
292       0.844257  0.534863  0.908334  0.652571  0.452429  0.443698
475       4.061927  4.872419  4.664161  3.245715  2.523616  3.463620
592       0.912714  0.650541  0.997030  0.704696  0.523500  0.539377
94        0.000000  0.000000  0.000000  0.000000  0.000000  0.000000
771       0.982099  0.740314  1.073185  0.779558  0.557920  0.591795

raceId         289       290       291       292       293       294  \
driverId
292       0.774599  0.816864  0.916301  0.905064  0.690120  0.450870
475       1.370115  0.032821  0.993232  1.408044  0.146316  1.561823
592       0.748921  0.743371  0.863746  0.868085  0.635048  0.470502
94        0.000000  0.000000  0.000000  1.000000  1.000000  0.000000
771       0.784096  0.764253  0.901160  0.918881  0.640773  0.496329

raceId         295       296       297       298       299       300  \
driverId
292       0.969639  0.835877  0.825350  0.599386  0.466092  0.788151
475       2.382719 -0.335133  0.480865  1.291747  0.863658  2.758608
592       0.981826  0.756488  0.765054  0.586742  0.474760  0.808422
94        0.000000  0.000000  0.000000  0.000000  0.000000  0.000000
771       1.041314  0.757551  0.790670  0.620965  0.479119  0.884947
```

```
raceId         301       302       303       304       305       306  \
driverId
292       0.763098  0.723852  1.208414  1.197097  0.507604  0.582451
475       5.447701 -0.183959  2.636881  4.154793  4.687691  3.879077
592       0.895890  0.664117  1.212861  1.250467  0.636301  0.681407
94        0.000000  0.000000  0.000000  0.000000  0.000000  0.000000
771       0.984656  0.664547  1.257418  1.338764  0.711888  0.738804

raceId         307       308       309       310       311       312  \
driverId
292       0.542552  0.548351  0.803782  0.696694  0.644537  0.695428
475       2.575489  2.478938  3.597481  0.726535  1.129336  1.107582
592       0.613027  0.595850  0.856542  0.667591  0.621746  0.666871
94        3.000000  0.000000  0.000000  0.000000  0.000000  0.000000
771       0.639020  0.635316  0.921738  0.684333  0.663223  0.707137

raceId         313       314       315       316       317       318  \
driverId
292       0.840370  0.581082  0.604474  0.506462  0.706566  0.623902
475      -0.282831  2.150856  4.004974  1.123492  2.922867  0.379219
592       0.755830  0.615494  0.707624  0.497982  0.761029  0.578910
94        0.000000  0.000000  0.000000  0.000000  3.000000  0.000000
771       0.768805  0.646997  0.760627  0.528999  0.802929  0.604144

raceId         319       320       321       322       323       324  \
driverId
292       0.678360  0.257280  0.622972  0.677089  0.789859  0.694438
475       2.318867  1.240113  3.893677  2.492245  1.827497  2.947935
592       0.721025  0.280981  0.725128  0.716094  0.795212  0.768705
94        0.000000  0.000000  0.000000  0.000000  0.000000  0.000000
771       0.754693  0.297393  0.774769  0.772782  0.834016  0.800485

raceId         325       326       327       328       329       330  \
driverId
292       0.590347  0.715671  0.688150  0.683549  0.664757  0.649327
475       4.012520  1.287614  3.473902  3.199356  2.706760  3.231439
592       0.680857  0.698440  0.761203  0.752355  0.732773  0.722336
94        0.000000  0.000000  0.000000  0.000000  0.000000  0.000000
771       0.747674  0.747209  0.830354  0.814138  0.766917  0.766421

raceId         331       332       333       334       335       336  \
driverId
292       0.660094  0.646368  0.608033  0.707001  0.749203  0.723536
475       3.705704  2.769662  1.746722  0.931721  3.600318  3.332477
592       0.755624  0.707477  0.614138  0.680651  0.803204  0.754380
94        0.000000  0.000000  0.000000  0.000000  0.000000  0.000000
771       0.813637  0.754104  0.657640  0.722029  0.868790  0.835126
```

```
raceId         337        338       339       340       341       342  \
driverId
292       5.366908   4.538896  2.887378  4.621794  3.240461  2.765328
475      10.443079  10.690882  1.024504  7.428476  9.310629  2.234915
592       5.434588   4.654449  2.805426  4.631480  3.422951  2.741937
94        4.258396   3.378339  2.728648  3.794123  2.252733  2.681388
771       5.589736   4.823972  2.791301  4.741446  3.545678  2.747736

raceId        343       344       345       346        347       348  \
driverId
292       3.820606  4.593097  5.339811  3.230824   4.962568  2.759503
475       4.268626  5.503382  6.381233  1.570359  10.284758  6.608871
592       3.766160  4.552255  5.270212  3.119848   5.068687  2.882333
94        3.193169  4.171544  4.838177  2.636251   4.157130  2.215450
771       3.842937  4.621511  5.339481  3.146728   5.204976  2.955037

raceId        349        350        351       352        353       354  \
driverId
292       2.822605   4.584574   4.242012  4.227160   5.255142  3.903331
475      -0.169324  12.703785  10.710514  7.785535  11.309295  5.750564
592       2.679510   4.788068   4.404249  4.298336   5.337141  3.927732
94        2.770573   3.269319   3.041150  3.536939   4.037459  3.352840
771       2.682728   4.968902   4.546761  4.384695   5.523644  3.985361

raceId        355       356       357       358       359       360  \
driverId
292       4.384551  0.600836  0.651485  0.719626  0.714614  0.650046
475       4.024105 -0.066976  3.050882  3.167633  2.507364  1.632688
592       4.306587  0.541799  0.731835  0.799920  0.772701  0.661504
94        4.081249  0.000000  0.000000  0.000000  0.000000  0.000000
771       4.340044  0.566461  0.776114  0.845738  0.801472  0.704907

raceId        361       362       363       364       365       366  \
driverId
292       0.657418  0.680614  0.618668  0.623317  0.622456  0.646440
475       1.802451  1.142992  1.693306  3.239698  1.361844  2.604328
592       0.671957  0.659247  0.616075  0.692929  0.618696  0.701757
94        0.000000  0.000000  2.000000  0.000000  0.000000  0.000000
771       0.702602  0.705331  0.672519  0.751136  0.652151  0.746206

raceId        367       368       369       370       371       372  \
driverId
292       0.705131  0.719614  0.817662  0.619437  0.658584  0.755552
475       1.962957  1.405706  2.605378  1.398294  0.568204  2.983176
592       0.726286  0.724071  0.856890  0.623711  0.643869  0.799818
94        0.000000  2.000000  0.000000  0.409332  1.000000  0.168033
```

```
771       0.774541  0.758110  0.902077  0.648254  0.649490  0.868291

raceId         373       374       375       376       377       378  \
driverId
292       0.687203  0.825871  0.768381  0.676397  0.691271  0.764679
475       4.568082  1.595751  2.818020  4.233335  3.179158  3.467643
592       0.804530  0.831564  0.827639  0.786573  0.773325  0.841302
94        0.120172  0.376913  0.255993  0.166703  1.000000  0.000000
771       0.878070  0.872634  0.879398  0.852829  0.819198  0.905159

raceId         379       380       381       382       383       384  \
driverId
292       0.673606  0.780711  0.735275  0.667464  1.067391  0.755447
475       2.173488  3.140693  3.263987  4.055963 -0.349904  1.849177
592       0.705479  0.857222  0.817185  0.770851  0.981055  0.776909
94        0.000000  0.000000  0.000000  0.000000  0.000000  0.000000
771       0.737310  0.903138  0.865079  0.835590  0.987126  0.818940

raceId         385       386       387       388       389       390  \
driverId
292       0.692023  0.740589  0.660812  0.841124  0.784041  0.809321
475       1.477245  3.266841  4.314061  3.891524  0.621592  1.791503
592       0.689115  0.822295  0.762711  0.901399  0.751309  0.819048
94        0.000000  0.000000  0.000000  0.147910  0.323780  0.384402
771       0.735089  0.870409  0.843207  0.988704  0.765623  0.864357

raceId         391       392       393       394       395       396  \
driverId
292       0.857838  0.677830  0.590484  0.655906  0.809613  0.691986
475       4.449904  4.726484  2.529428  2.239594  4.678571  5.456648
592       0.961206  0.790352  0.607024  0.663383  0.903555  0.820020
94        0.461221  0.039923 -0.186291  0.021932  0.374306  0.070184
771       1.022009  0.869215  0.672118  0.713892  0.980050  0.914028

raceId         397       398       399       400       401       402  \
driverId
292       0.651594  0.687882  0.747119  0.625549  0.695238  0.997820
475       2.057269  4.464560  3.146855  1.614076  1.494890  2.085219
592       0.649366  0.775599  0.795133  0.614787  0.666003  1.024706
94       -0.003410  0.069823  0.143537 -0.121059  0.075136  0.775580
771       0.701798  0.853411  0.867034  0.667694  0.711926  1.046604

raceId         403       404       405       406       407       408  \
driverId
292       1.092152  0.559158  0.603414  0.617293  0.771654  0.785589
475       0.020380  4.270443  2.028889  3.498345  2.155169  1.039950
592       1.024198  0.654509  0.634711  0.677796  0.794640  0.767869
```

94	0.770807	0.111026	0.036943	-0.039421	0.182294	0.303713
771	1.030091	0.724877	0.672022	0.758762	0.845316	0.788819

raceId	409	410	411	412	413	414 \
driverId						
292	0.626741	0.729929	0.686257	0.616780	0.569549	0.526948
475	2.172295	2.515622	1.912392	2.188751	4.043886	3.932926
592	0.632798	0.784688	0.677626	0.616562	0.645872	0.603646
94	-0.146588	0.345366	-0.054903	-0.102273	-0.065844	-0.067148
771	0.694516	0.818314	0.737209	0.679619	0.722318	0.675183

raceId	415	416	417	418	419	420 \
driverId						
292	0.882914	0.664383	0.568928	0.687177	0.568612	1.020710
475	1.934097	2.979335	2.229683	2.788045	3.455203	1.795384
592	0.899906	0.687433	0.583926	0.733501	0.631154	1.025246
94	0.499150	0.050381	-0.192925	0.109816	-0.046269	0.000000
771	0.943129	0.755143	0.644035	0.790294	0.710484	1.066582

raceId	421	422	423	424	425	426 \
driverId						
292	0.792983	0.906320	1.038053	1.220448	1.271700	0.862537
475	1.458654	0.823664	1.733710	0.995181	1.122397	4.197716
592	0.813296	0.880127	1.040435	1.187025	1.235560	0.926275
94	0.000000	0.000000	0.657468	0.000000	0.000000	0.000000
771	0.824167	0.894716	1.079945	1.208108	1.253457	1.012192

raceId	427	428	429	430	431	432 \
driverId						
292	0.984763	1.036445	0.750040	0.507986	0.496219	0.507823
475	2.839737	1.430511	2.579747	2.138672	3.840175	2.568842
592	1.020141	1.030984	0.804584	0.552637	0.583299	0.557795
94	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
771	1.078907	1.058501	0.851292	0.592953	0.663197	0.605826

raceId	433	434	435	436	437	438 \
driverId						
292	0.611503	0.996937	1.221652	0.950056	0.601027	1.221409
475	1.219253	0.212923	0.852540	1.829258	1.930605	0.669043
592	0.605604	0.927082	1.182520	0.953643	0.627135	1.182811
94	0.000000	0.000000	0.000000	0.486630	0.121803	1.143125
771	0.635821	0.951155	1.192511	0.999858	0.668238	1.188748

raceId	439	440	441	442	443	444 \
driverId						
292	0.977789	0.726441	0.427847	0.586484	0.699052	1.227400
475	2.068172	0.623082	1.405566	4.316095	4.121826	1.016225

```
592       0.989635  0.692595  0.454194  0.667196  0.773715  1.188575
94        0.549404  0.344684  0.199019 -0.069732  0.225682  1.061239
771       1.037988  0.711549  0.480622  0.756112  0.852839  1.205849

raceId         445       446       447       448       449       450  \
driverId
292       0.644389  0.596516  0.573602  0.576821  0.809807  0.751809
475       1.615783  1.511404  2.862306  1.452813  2.801914  3.024074
592       0.667691  0.601613  0.620547  0.578892  0.876155  0.799110
94        0.388084  0.000113  0.170280 -0.051715  0.000000  0.191173
771       0.687685  0.647596  0.671801  0.625257  0.903776  0.866276

raceId         451       452       453       454       455       456  \
driverId
292       0.566319  0.552706  0.943495  0.725172  0.959206  0.828865
475       2.855291  4.702940  1.984523  3.532052  1.002063  2.661119
592       0.631103  0.664046  0.968558  0.781272  0.928597  0.846644
94        0.023148  0.113899  0.735447  0.117951  0.628286  0.371766
771       0.687573  0.739787  0.988440  0.861967  0.962009  0.908393

raceId         457       458       459       460       461       462  \
driverId
292       0.788566  1.123312  1.048797  0.600734  0.969012  0.713823
475       2.516598  1.313714  0.207061  2.989901  0.747587  2.544588
592       0.813968  1.099561  0.981979  0.637236  0.929474  0.734941
94        0.265114  0.938679  0.883595 -0.088570  0.785251  0.118681
771       0.876234  1.121334  0.999622  0.715951  0.951740  0.803515

raceId         463       464       465       466       467       468  \
driverId
292       1.043965  0.789143  0.500801  0.832450  0.801697  0.811649
475       0.048983  2.970573  3.056018  1.594480  3.869513  1.480262
592       0.978159  0.810892  0.536087  0.820775  0.889344  0.808756
94        0.978447  0.437304 -0.147852  0.594396  0.259169  0.311066
771       0.985674  0.875456  0.615199  0.853712  0.957721  0.849985

raceId         469       470       471       472       473       474  \
driverId
292       0.761616  1.961241  0.873827  0.748902  1.031683  0.743757
475       2.821616  4.188713  1.033082  0.357728  0.859480  3.449619
592       0.824572  2.017677  0.851425  0.708604  1.006109  0.791094
94        0.411717  1.772289  0.707419  0.580256  0.957441  0.363267
771       0.857170  2.057482  0.871552  0.718529  1.015231  0.856027

raceId         475       476       477       478       479       480  \
driverId
292       0.773264  0.841564  0.998357  0.867508  1.028988  0.797213
```

```
475       3.275927  3.527561  1.955093 -1.513319  1.968784  3.208201
592       0.825510  0.930204  1.007426  0.750897  1.044680  0.850568
94        0.412852  0.482787  0.771785  0.952825  0.846584  0.247692
771       0.880510  0.970843  1.047539  0.742506  1.067858  0.913438

raceId         481       482       483       484       485       486  \
driverId
292       1.066254  1.011566  0.683131  0.772936  0.478162  0.496812
475       0.308753  0.834629  4.309421  4.157916  4.885783  3.239125
592       1.010901  0.989741  0.794699  0.887409  0.594546  0.551097
94        0.987717  0.865004  0.344292  0.446149 -0.101257  0.084420
771       1.021305  1.002439  0.852645  0.931946  0.684119  0.613434

raceId         487       488       489       490       491       492  \
driverId
292       0.842480  0.894935  0.955861  0.674599  0.845693  0.503366
475       3.552492  1.765255  1.772859  2.741071  2.242906  3.536746
592       0.929571  0.925706  0.982220  0.714179  0.884832  0.568746
94        0.413240  0.889874  0.844053  0.166446  0.646831 -0.050235
771       0.966617  0.935158  0.991190  0.778484  0.906665  0.645989

raceId         493       494       495       496       497       498  \
driverId
292       0.838336  0.525324  0.691516  0.866519  0.647772  0.511072
475       2.412607  4.053540  4.794073  1.488114  3.596980  2.877621
592       0.865731  0.617273  0.823668  0.877288  0.733113  0.570380
94        0.672689 -0.103988  0.082940  0.766886  0.122310  0.310715
771       0.902517  0.701235  0.900702  0.889877  0.794144  0.618035

raceId         499       500       501       502       503       504  \
driverId
292       0.626109  0.725841  0.728025  0.701711  0.774324  0.605330
475       0.570635  1.116073  3.554170  1.951340  3.673297  2.725910
592       0.616289  0.720050  0.764380  0.744803  0.849042  0.676077
94        0.883297  0.877569 -0.192549  0.910418  0.356249  0.769251
771       0.621127  0.734618  0.854126  0.758828  0.902593  0.704323

raceId         505       506       507       508       509       510  \
driverId
292       0.488171  0.807233  0.845945  0.564937  0.558036  0.712004
475       3.993898  3.704265  2.653087  2.925910  4.184518  3.106588
592       0.592022  0.895562  0.890778  0.606983  0.649186  0.800300
94        0.253887  0.603392  0.657984  0.259832  0.154618  0.815755
771       0.651405  0.939574  0.924062  0.664770  0.720531  0.826586

raceId         511       512       513       514       515       516  \
driverId
```

```
292       0.728899  0.905750  0.916816  0.856814  0.840763  0.737749
475       3.444972  1.522083  1.524575  1.729643  2.170774  2.307217
592       0.826170  0.918358  0.935017  0.887871  0.893679  0.788902
94        0.755565  0.946000  1.009181  0.744197  0.800141  0.572322
771       0.858651  0.925396  0.938626  0.905931  0.915324  0.816442

raceId         517       518       519       520       521       522  \
driverId
292       0.690345  0.645563  0.903349  0.866014  0.665041  0.747867
475       2.904718  3.328636 -0.051371  0.926961  2.853898  2.448160
592       0.768679  0.738095  0.862272  0.850835  0.748956  0.821463
94        0.502676  0.303920  1.101171  0.744585  0.679387  0.931379
771       0.807054  0.789670  0.847271  0.865718  0.776791  0.835246

raceId         523       524       525       526       527       528  \
driverId
292       0.488416  0.720940  0.657435  0.727646  0.789933  0.617434
475       3.946638  2.981339  2.576728 -1.027366  2.074724  3.964652
592       0.613959  0.802746  0.737631  0.669285  0.809688  0.690300
94        0.375690  0.318547  0.767518  1.258729  0.202037 -0.036254
771       0.666082  0.848721  0.757752  0.634870  0.851829  0.770970

raceId         529       530       531       532       533       534  \
driverId
292       0.609152  0.625982  0.699925  0.652268  0.669410  0.659765
475       2.443740  2.994145  1.893430  1.936290  1.878025  2.854796
592       0.670218  0.690238  0.735860  0.692211  0.707609  0.703777
94        0.604912  0.437960  0.250852  0.600672  0.503565 -0.010078
771       0.697235  0.731820  0.765601  0.713133  0.734118  0.760489

raceId         535       536       537       538       539       540  \
driverId
292       0.690291  0.567422  0.722115  0.676210  0.672840  0.643815
475       2.323053  3.094957  1.898086  1.768810  2.005272  2.073156
592       0.728828  0.632498  0.752702  0.711388  0.707206  0.678368
94        0.403870  0.121362  0.357704  0.654766  0.317556  0.170100
771       0.770133  0.679938  0.790675  0.731507  0.742025  0.708577

raceId         541       542       543       544       545       546  \
driverId
292       0.510037  0.812166  0.753658  0.772895  0.580034  0.822160
475       5.279709  1.929700  3.498221  4.788420  3.141604  2.049352
592       0.656641  0.851897  0.834597  0.847725  0.650213  0.850416
94        0.210927  0.748712  0.316699 -0.023509 -0.259580  0.206276
771       0.738821  0.870387  0.898516  0.946944  0.714253  0.892525

raceId         547       548       549       550       551       552  \
```

```
driverId
292       0.660206  0.457223  0.600209  0.822894  0.857193  0.653618
475       2.049978  3.700877  1.946548  0.843239  0.991791  2.282961
592       0.697756  0.565915  0.631275  0.813435  0.835534  0.662712
94        0.385935 -0.193144  0.276520  0.860727  0.503708 -0.084231
771       0.729923  0.630411  0.660737  0.812236  0.857851  0.720681

raceId         553       554       555       556       557       558  \
driverId
292       0.483652  0.525071  0.544990  0.580802  0.753630  0.680448
475       3.257402  3.760846  3.131719  2.150123  3.187754  3.053694
592       0.568461  0.620525  0.615535  0.625006  0.792786  0.762691
94       -0.233606  0.160363 -0.161901  0.141964 -0.061270  0.095566
771       0.627286  0.675975  0.676126  0.660651  0.869585  0.813745

raceId         559       560       561       562       563       564  \
driverId
292       0.868355  0.756576  0.702181  0.776893  0.710049  0.581261
475       5.050161  1.656303  2.753690  1.960193  3.217705  3.082110
592       0.954971  0.773581  0.732581  0.796030  0.737234  0.653423
94       -0.082127  0.182992 -0.224330  0.280218 -0.135455 -0.055380
771       1.053818  0.801800  0.799395  0.831352  0.814331  0.706544

raceId         565       566       567       568       569       570  \
driverId
292       0.662489  0.577881  0.875155  0.539135  0.801562  1.116996
475       2.762787  2.815717  1.484827  3.006217  2.289891  0.509402
592       0.724629  0.654088  0.854680  0.614414  0.811573  1.069364
94       -0.097335  0.040750  0.401638 -0.152376  0.190074  1.011026
771       0.777819  0.703300  0.895043  0.670134  0.870959  1.073352

raceId         571       572       573       574       575       576  \
driverId
292       0.965214  0.565358  0.840912  0.687239  0.917294  0.926483
475       3.715957  3.310760  2.138449  3.248689  1.716060  3.683743
592       1.002429  0.651624  0.846913  0.733181  0.919921  0.947480
94       -0.130481  0.282536  0.259930 -0.155817  0.618186  0.071200
771       1.090762  0.701136  0.899988  0.810423  0.950870  1.045612

raceId         577       578       579       580       581       582  \
driverId
292       0.964422  0.000000  0.496787  0.574759  0.545319  0.638307
475       1.251953  3.986814  0.225688  3.359761  3.891742  3.500808
592       0.928721  0.618495  0.474548  0.632755  0.648412  0.708443
94        0.463812  0.079336  0.472709 -0.257570 -0.240097 -0.046315
771       0.970815  0.679544  0.475837  0.710777  0.717575  0.769523
```

```
raceId         583       584       585       586       587       588  \
driverId
292       0.792728  0.786790  0.838878  0.643251  0.410744  0.750402
475       3.541837  2.973118  2.413431  3.256820  0.381630  3.592675
592       0.826384  0.806927  0.841814  0.710706  0.391162  0.795142
94       -0.112047 -0.221630  0.167209  0.376780  0.256739 -0.033728
771       0.909656  0.884610  0.915407  0.754101  0.402330  0.879740

raceId         589       590       591       592       593       594  \
driverId
292       0.708339  0.672153  0.931737  0.000000  0.526161  0.589457
475       3.375633  2.934824  1.399337  2.212960  3.270087  3.350902
592       0.747367  0.711693  0.904065  0.749078  0.587731  0.642759
94       -0.313668  0.134362  0.470677  0.729836 -0.358750 -0.379549
771       0.834483  0.770710  0.950527  0.768234  0.664183  0.732441

raceId         595       596       597       598       599       600  \
driverId
292       0.711424  0.751328  0.567659  0.513002  0.694043  0.595006
475       3.482588  2.506312  3.170215  4.608686  3.480050  3.791899
592       0.806502  0.796165  0.626385  0.644545  0.767847  0.700200
94        0.485487  0.191252 -0.299317  0.020589 -0.011724  0.285450
771       0.857559  0.850792  0.700024  0.718668  0.845203  0.757973

raceId         601       602       603       604       605       606  \
driverId
292       0.538737  0.676855  0.740163  0.611205  1.018526  0.902812
475       3.845580  3.407556  2.792675  3.049420  1.997372  2.186745
592       0.612163  0.751046  0.762102  0.659336  1.002445  0.891982
94        0.196650  0.297444  0.124524  0.249598  0.405511  0.233024
771       0.679991  0.816207  0.836081  0.716891  1.064623  0.961991

raceId         607       608       609       610       611       612  \
driverId
292       0.000000  1.151984  0.638772  0.614449  0.857300  0.638754
475       2.096542  0.819663  2.589032  3.436929  4.515325  4.324614
592       0.594199  1.092873  0.679068  0.691770  0.948073  0.752678
94       -0.086236  0.722780 -0.064106  0.026141  0.525572  0.410045
771       0.660610  1.129537  0.743361  0.765456  1.022346  0.815371

raceId         613       614       615       616       617       618  \
driverId
292       0.694666  0.562282  0.736980  0.727112  0.790289  0.938716
475       2.858101  2.732742  2.623096  4.772018  3.529559  0.557660
592       0.738959  0.612990  0.780591  0.864026  0.871103  0.890202
94        0.485518 -0.056234  0.175884  0.391221  0.504214  0.658867
771       0.789126  0.675736  0.837883  0.931616  0.930986  0.914832
```

```
raceId         619       620       621       622       623       624  \
driverId
292       0.646766  0.703921  0.965837  1.023422  0.989183  1.072882
475       4.087451  3.493554  2.057072  0.177781  1.130700  2.907162
592       0.732285  0.763128  0.945471  0.948141  0.961133  1.075336
94        0.389060  0.110172  0.538653  0.677970  0.655384  0.215398
771       0.800118  0.837494  1.001727  0.975636  0.992859  1.161055

raceId         625       626       627       628       629       630  \
driverId
292       0.762029  0.928567  1.021854  1.080991  1.003885  0.604366
475       2.868175  1.952596  1.826575  1.116006  0.622805  3.783251
592       0.801138  0.913070  1.033927  1.018709  0.931774  0.682527
94        0.081755  0.271525  0.959128  0.622111  0.577990  0.014610
771       0.873541  0.977743  1.056362  1.066599  0.969505  0.759352

raceId         631       632       633       634       635       636  \
driverId
292       0.389692  0.841209  0.856836  0.742037  1.211543  0.749911
475       3.580741  2.417662  3.548075  5.732254  1.257042  2.899243
592       0.465230  0.878959  0.912326  0.880689  1.183299  0.790559
94       -0.235428  0.315837  0.163325  0.023000  1.160557 -0.093604
771       0.543647  0.923818  0.985643  0.985070  1.207362  0.862573

raceId         637       638       639       640       641       642  \
driverId
292       0.760517  0.716605  1.270723  0.988398  0.549060  0.701023
475       4.510627  4.925452 -1.562743  3.158545  5.980992  2.617975
592       0.866978  0.840923  1.139359  1.048969  0.713392  0.760694
94        0.098330 -0.116954  1.079905  0.913592 -0.170624  0.417617
771       0.950820  0.929363  1.127536  1.086878  0.815605  0.798232

raceId         643       644       645       646       647       648  \
driverId
292       0.919581  0.939850  1.126030  1.425949  0.974024  1.066216
475       4.733098  3.779654  1.150037 -0.606072  2.758838  0.613115
592       0.995795  1.004931  1.078765  1.331050  0.995814  1.008907
94        0.296222  0.009276  0.882733  1.374690  0.405756  0.912212
771       1.091667  1.082797  1.111115  1.320358  1.057170  1.032345

raceId         649       650       651       652       653       654  \
driverId
292       0.953479  0.927200  1.095623  0.766066  1.329502  0.971013
475       1.923237  0.583593  1.270383  3.761178  0.330595 -2.124341
592       0.946457  0.879148  1.083741  0.841587  1.266847  0.834923
94        0.766432  0.811533  1.032321  0.123158  1.250188  0.911910
```

```
771        0.983571  0.897579  1.095770  0.909269  1.274057  0.820865

raceId          655       656       657       658       659       660  \
driverId
292        1.361229  1.101899  1.304208  1.571474  1.250068  1.507943
475        2.686833  4.694249  5.077268  0.435138  3.556130  4.947858
592        1.379218  1.175275  1.393417  1.496382  1.286323  1.583276
94         1.254502  0.065812  0.389415  1.253343  0.440731  0.627569
771        1.413276  1.265602  1.488371  1.494163  1.362905  1.675668

raceId          661       662       663       664       665       666  \
driverId
292        1.313041  1.570011  1.267832  1.222708  1.291600  1.225203
475        4.551713  3.380474  3.810155  1.773693  0.963108  4.500520
592        1.378788  1.582971  1.321484  1.205206  1.255372  1.282421
94         0.466533  0.873391  0.485700  1.084838  1.309264  0.720259
771        1.474393  1.650925  1.396247  1.241961  1.266231  1.367993

raceId          667       668       669       670       671       672  \
driverId
292        0.832404  1.668383  1.342050  1.303096  1.045454  1.259387
475        3.555233  4.613730  2.556865  0.953516  3.643392  3.579141
592        0.894909  1.709556  1.341713  1.264200  1.106193  1.309441
94         0.337579  0.747870  0.731682  0.945090  0.320814  0.894294
771        0.947433  1.786613  1.380460  1.287890  1.176770  1.364878

raceId          673       674       675       676       677       678  \
driverId
292        1.137629  1.080293  1.027760  1.160058  1.045848  1.252358
475        0.576457  4.337587  2.828615  2.683236  4.375128  0.243191
592        1.090211  1.147289  1.052001  1.165553  1.123053  1.186344
94         0.987850  0.152662  0.814362  0.717540  0.112060  0.911332
771        1.102073  1.233206  1.097136  1.215243  1.200930  1.185960

raceId          679       680       681       682       683       684  \
driverId
292        1.438725  1.299450  0.961844  1.242912  1.449806  0.892132
475        1.174007  1.511956  4.882833  0.951610  3.050429  4.769753
592        1.400491  1.256156  1.054819  1.204489  1.446145  0.986044
94         1.362579  0.982379  0.495953  0.777128  1.032504  0.447385
771        1.415007  1.283218  1.139320  1.231989  1.514682  1.065746

raceId          685       686       687       688       689       690  \
driverId
292        0.959796  1.484393  1.141076  0.980915  0.954331  1.109881
475        2.729907  2.555209  3.690248  3.954972  5.232744  0.650300
592        0.974306  1.463091  1.202186  1.043577  1.061762  1.063272
```

```
94        0.782602  1.207165  0.966439  0.438094  0.529718  0.525361
771       1.026343  1.513989  1.254108  1.104298  1.148180  1.087687

raceId         691       692       693       694       695       696  \
driverId
292       1.402700  1.562290  1.231314  1.245204  1.349745  1.076481
475       2.065613  6.357414  3.526746  2.871260  2.286313  1.188549
592       1.400351  1.656542  1.249582  1.250309  1.336919  1.053471
94        1.386744  0.535293  0.820974  0.682193  1.072276  0.973572
771       1.423846  1.790990  1.320022  1.310806  1.383137  1.070967

raceId         697       698       699       700       701       702  \
driverId
292       0.881583  1.347313  0.607334  1.445031  0.777845  0.852437
475       3.817083  3.043194  4.176164  2.317420  4.367281  4.847257
592       0.964322  1.368437  0.710019  1.437547  0.863993  0.955621
94        0.540080  1.243541  0.160212  0.807118  0.007234  0.157392
771       1.018040  1.412280  0.767228  1.470848  0.949699  1.039683

raceId         703       704       705       706       707       708  \
driverId
292       0.731518  0.858562  0.925450  0.900823  1.577341  1.334887
475       3.680842  4.051253  3.449186  0.235642  1.024911  1.332503
592       0.812002  0.935076  0.981699  0.850359  1.511145  1.311090
94        0.216512  0.111894  0.417339  0.283701  1.254429  1.222360
771       0.861836  1.008550  1.032537  0.867697  1.524572  1.323880

raceId         709       710       711       712       713       714  \
driverId
292       1.073142  1.016365  1.052321  1.486870  0.927799  1.150183
475       1.813851  4.542096  4.800002  0.977466  4.709027  0.898433
592       1.065768  1.107477  1.133965  1.426767  1.016645  1.120949
94        0.617507  0.298956  0.057265  1.125362  0.087850  1.106958
771       1.089511  1.181250  1.235353  1.444746  1.101118  1.116550

raceId         715       716       717       718       719       720  \
driverId
292       1.187861  1.063638  1.343092  1.253892  1.112799  0.890606
475       0.703594  1.747469  0.814322  1.667348  0.885861  4.124736
592       1.152348  1.069857  1.295561  1.248877  1.074276  0.970265
94        1.145688  0.992777  1.186333  1.052597  0.826196 -0.025299
771       1.155465  1.087744  1.291520  1.266394  1.080719  1.055962

raceId         721       722       723       724       725       726  \
driverId
292       0.981272  0.700888  0.904888  0.758059  0.721219  1.037905
475       4.417879  4.275589  4.660744  4.059740  4.602994  2.420875
```

```
592        1.071173  0.789692  1.003866  0.853269  0.819709  1.042671
94         0.233201 -0.105307  0.165991  0.369501 -0.144307  0.456673
771        1.149482  0.878329  1.079705  0.912605  0.914932  1.085923

raceId         727       728       729       730       731       732  \
driverId
292        0.861448  0.886682  1.191008  1.049771  0.827931  1.221184
475        4.977984  3.563834  0.236867  1.309631  4.593893  0.936958
592        0.954388  0.945796  1.128339  1.034170  0.919862  1.187214
94        -0.048165 -0.037657  1.026214  0.815965 -0.023286  1.021724
771        1.056995  1.017593  1.121510  1.057400  1.002621  1.201591

raceId         733       734       735       736       737       738  \
driverId
292        0.889761  0.875052  0.824908  1.067117  1.323752  0.978645
475        4.770740  1.639897 -0.035501  4.259405  0.341286  9.000000
592        0.987035  0.875025  0.774800  1.128514  1.246201  1.022406
94         0.048141  0.439542  0.566519  0.045291  0.895005  0.697812
771        1.075294  0.894982  0.766449  1.217272  1.258395  1.069388

raceId         739       740       741       742       743       744  \
driverId
292        1.064199  0.882871  0.762438  0.724154  0.531126  0.678269
475        3.000000  0.000000  0.000000  0.000000  9.000000  0.000000
592        1.122064  0.901951  0.751850  0.744863  0.628425  0.663375
94         0.617847  0.828237  0.370063  0.612194  0.098546  0.540586
771        1.187074  0.929626  0.774441  0.781462  0.695054  0.679282

raceId         745       746       747       748       749       750  \
driverId
292        1.362496  0.898525  0.883107  0.714739  1.120596  1.140027
475        0.000000  0.000000  8.000000  2.213715  3.000000  0.000000
592        1.290121  0.862490  1.009785  0.740625  1.101155  1.129112
94         1.189842  0.627893 -0.144584  0.363277  0.848466  0.719843
771        1.291120  0.876140  1.122236  0.782580  1.143080  1.187778

raceId         751       752       753       754       755       756  \
driverId
292        0.943166  1.003077  1.292279  1.477574  0.903486  1.071282
475        2.403377  2.212224  0.000000  5.128975  8.000000  0.000000
592        0.949080  1.004772  1.288871  1.549306  0.920882  1.059041
94         0.500344  0.832522  0.702424  0.826376  0.590743  0.737234
771        1.010240  1.050805  1.358829  1.643589  0.967303  1.116177

raceId         757       758       759       760       761       762  \
driverId
292        0.715813  1.410928  1.115746  0.892969  1.594267  1.294349
```

```
475       2.058676  1.000000  1.000000  6.500000  0.000000  9.000000
592       0.737301  1.373881  1.144317  1.018497  1.530292  1.393306
94        0.394078  1.287717  0.696286 -0.173567  1.601742  0.460056
771       0.775925  1.400460  1.207727  1.154285  1.530421  1.484785

raceId         763       764       765       766       767       768  \
driverId
292       1.078137  1.256511  2.070227  0.833142  1.157997  0.708649
475       8.000000  0.000000  8.000000  0.000000  9.000000  2.001942
592       1.234156  1.192588  2.127301  0.792535  1.269826  0.000000
94        0.039228  0.964312  1.303681  0.781550  0.323670  0.373380
771       1.369417  1.212679  2.226387  0.803884  1.365520  0.766469

raceId         769       770       771       772       773       774  \
driverId
292       1.231466  0.775987  1.160154  0.857136  1.303867  0.991053
475       0.000000  6.000000  0.000000  1.000000  8.000000  0.000000
592       1.231093  0.872658  1.147092  0.816956  1.442990  0.999185
94        0.881289  0.209242  0.953254  0.949365  0.172678  0.742822
771       1.267773  0.948537  1.169874  0.811768  1.567506  1.033047

raceId         775       776       777       778       779       780  \
driverId
292       0.708132  1.113935  0.983586  0.711694  1.381572  1.026794
475       9.000000  1.000000  0.000000  2.762620  4.408575  2.500000
592       0.885318  1.108748  0.996201  0.000000  1.442512  1.023283
94       -0.300455  0.491189  0.276254  0.252626  0.538212  0.841788
771       1.009821  1.145360  1.044075  0.806855  1.521088  1.046823

raceId         781       782       783       784       785       786  \
driverId
292       0.800711  1.223862  0.917573  1.378599  0.902675  0.690098
475       2.000000  9.000000  8.000000  0.000000  8.000000  1.543299
592       0.881653  1.322469  1.022011  1.342902  0.974891  0.000000
94       -0.075670  0.426898  0.160101  1.074554  0.396652  0.432578
771       0.962644  1.410171  1.113982  1.365695  1.042812  0.726428

raceId         787       788       789       790       791       792  \
driverId
292       1.582358  1.189454  0.591468  0.901850  0.760176  0.394456
475       1.500000  0.500000  1.000000  6.000000  9.000000  0.500000
592       1.541802  1.164420  0.612511  1.006936  0.838382  0.410671
94        1.425713  0.819933  0.009778 -0.052083  0.348533 -0.203263
771       1.561958  1.187737  0.649679  1.105219  0.904247  0.441990

raceId         793       794       795       796       797       798  \
driverId
```

```
292        0.963974  0.667232  0.771097  1.099216  0.607722  0.681864
475        0.000000  1.072351  6.000000  6.000000  9.000000  1.000000
592        0.924259  0.000000  0.865631  1.184051  0.728804  0.703103
94         0.800647  0.489401 -0.080458  0.240205 -0.149723  0.040874
771        0.936705  0.681657  0.952377  1.274815  0.815881  0.743188

raceId          799       800       801       802       803       804  \
driverId
292        0.842007  0.523768  1.097243  0.842510  0.554119  0.633494
475        4.717050  1.648736  4.000000  1.540113  0.000000  0.000000
592        0.936721  0.000000  1.119015  0.844354  0.582230  0.667660
94        -0.042838  0.245463  0.383529  0.284686  0.127235 -0.032971
771        1.030231  0.574806  1.173860  0.875835  0.624474  0.720226

raceId          805       806       807       808       809       810  \
driverId
292        0.909288  0.680346  0.893516  1.052521  0.574582  0.878794
475        0.000000  0.000000  0.000000  4.409565  1.486469  0.000000
592        0.953025  0.685858  0.905841  1.126853  0.586964  0.935282
94         0.100917  0.094216  0.437784  0.646247  0.325760  0.492592
771        1.018211  0.720195  0.948193  1.203534  0.614717  1.001889

raceId          811       812       813       814       815       816  \
driverId
292        1.020577  0.564217  0.426847  0.463374  0.750295  0.549220
475        3.288256  0.000000  4.885539  0.000000  5.155221  0.000000
592        1.062515  0.616546  0.550104  0.506436  0.862823  0.584819
94         0.727535 -0.104290 -0.332446 -0.118610  0.197188 -0.165148
771        1.118881  0.676060  0.644632  0.558160  0.957890  0.637098

raceId          817       818       819       820       821       822  \
driverId
292        1.139137  0.752334  0.729274  0.916160  0.527317  0.533066
475        0.000000  0.664835  0.000000  6.121274  0.000000  6.507200
592        1.116876  0.000000  0.825618  1.048117  0.617418  0.697099
94         0.772098  0.649712 -0.133998 -0.201231 -0.289961 -0.655579
771        1.140842  0.741438  0.920835  1.168711  0.699823  0.827169

raceId          823       824       825       826       827       828  \
driverId
292        0.887026  0.369004  0.713884  0.688204  1.629828  0.602509
475        0.000000  0.000000  0.000000  1.371187  4.758379  2.173158
592        0.983383  0.472279  0.717795  0.691617  1.682425  0.634732
94        -0.040821 -0.417901  0.053529  0.459262  1.101658  0.252707
771        1.083921  0.558701  0.752202  0.000000  1.767410  0.675409

raceId          829       830       831       832       833       834  \
```

```
driverId
292       0.818199  0.585289  0.721896  0.633457  1.011377  0.661811
475       2.574507  4.470791  4.397810  3.761533  4.272034  2.955703
592       0.851432  0.688127  0.813766  0.712744  1.081772  0.716506
94        0.227887 -0.174233  0.275898 -0.190122  0.535493 -0.037718
771       0.901729  0.773761  0.892636  0.788540  1.161410  0.774292

raceId         835       836       837       838       839       841   \
driverId
292       0.676233  1.294925  1.231087  0.912767  0.708619  4.183991
475       1.130270  2.629335  3.389405  3.079443  2.790181  4.301316
592       0.673509  1.301613  1.266067  0.956342  0.752292  4.141209
94        0.489056  0.964906  0.462857  0.172984  0.386126  4.119024
771       0.000000  1.349487  1.332392  1.017043  0.804551  4.168907

raceId         842       843       844       845       846       847   \
driverId
292       4.343833  4.253297  4.088117  4.649190  4.354845  3.640585
475       8.278460  2.681970  5.204434  6.026908  8.587025  8.674244
592       4.423168  4.144227  4.083311  4.631805  4.449846  3.761911
94        3.756069  4.078895  3.628831  4.141831  3.761499  2.697080
771       4.511230  4.165606  4.129693  4.695828  4.540930  3.874561

raceId         848       849       850       851       852        853   \
driverId
292       4.311331  4.504277  3.998830  4.403547  3.683452   5.285734
475       6.476421  8.148155  3.940314  8.605036  8.341236  10.188485
592       4.342343  4.568200  3.948091  4.494919  3.807919   5.364208
94        3.905312  3.775159  3.794435  3.839287  2.774073   4.482442
771       4.404705  4.670461  3.991728  4.595941  3.906144   5.490277

raceId         854        855       856       857       858       859   \
driverId
292       4.197889   4.574606  4.410197  4.430570  4.162721  3.042350
475       6.923591  10.815598  4.757133  9.259159  6.486780  6.961126
592       4.246650   4.727172  4.373469  4.536767  4.177655  3.166115
94        3.656761   3.665768  4.224910  3.469802  3.586719  2.408063
771       4.316654   4.870756  4.408983  4.650642  4.277469  3.242156

raceId         860        861       862       863       864       865   \
driverId
292       4.169225   4.611964  3.567076  3.685094  4.081628  2.859251
475       6.820635  10.471679  2.728462  4.775996  9.107001  3.355757
592       4.207661   4.711266  3.470947  3.661135  4.153612  2.854913
94        3.766500   3.415391  2.876458  2.988218  2.932419  2.313369
771       4.283676   4.877835  3.513171  3.708048  4.288797  2.895003
```

raceId	866	867	868	869	870	871	\
driverId							
292	5.099124	3.667856	2.874136	4.405784	5.333727	3.463968	
475	5.071253	12.265410	6.824604	13.451855	6.855106	7.744797	
592	4.984203	3.883588	3.000946	4.631150	5.274167	3.562468	
94	4.498198	1.921509	2.339651	3.017527	4.696303	2.761080	
771	5.055331	4.084423	3.077855	4.827847	5.361736	3.657637	

raceId	872	873	874	875	876	877	\
driverId							
292	6.180662	3.991308	4.413682	2.848562	3.776296	4.706244	
475	10.349033	7.950914	5.148753	5.160163	5.829385	11.947991	
592	6.156483	4.063657	4.362210	2.922804	3.823550	4.853461	
94	4.820450	3.223725	4.156644	2.640496	3.642735	3.453686	
771	6.334879	4.159398	4.409831	2.960537	3.870758	5.019182	

raceId	878	879	880	881	882	883	\
driverId							
292	5.415893	2.845873	5.283611	3.613779	5.861541	4.807935	
475	6.798025	9.524014	9.540038	1.818368	10.925430	6.158094	
592	5.367633	3.063101	5.321080	3.509953	5.909798	4.750189	
94	5.118793	2.219341	4.249498	3.427026	4.783031	4.058230	
771	5.444510	3.189300	5.450516	3.508558	6.065920	4.820010	

raceId	884	885	886	887	888	890	\
driverId							
292	4.954310	3.473017	3.872787	4.437799	4.796958	5.258731	
475	10.895209	3.168369	3.854283	7.792413	6.849899	6.747690	
592	5.053812	3.408467	3.828702	4.447130	4.780918	5.213062	
94	3.529195	2.715364	3.627347	2.767435	4.194665	4.673000	
771	5.218177	3.451727	3.858380	4.593262	4.853125	5.293913	

raceId	891	892	893	894	895	896	\
driverId							
292	4.096495	2.966719	5.608082	4.790768	2.672247	4.760478	
475	4.379861	3.516666	8.622718	6.077155	3.269922	5.098300	
592	4.060739	2.972254	5.594348	4.737921	2.680287	4.660167	
94	3.855741	2.807115	4.615857	4.167862	2.457574	3.909677	
771	4.097159	2.993648	5.707976	4.802387	2.692535	4.727074	

raceId	897	898	899	900	901	902	\
driverId							
292	3.024811	4.206877	4.692567	3.495805	4.533380	5.536478	
475	2.476184	4.195848	8.082601	6.586228	3.856850	5.722101	
592	2.978709	4.145813	4.734159	3.523549	4.448944	5.398996	
94	2.662567	3.897056	3.682741	1.858638	3.978801	4.311635	
771	2.991503	4.174835	4.836056	3.652016	4.497442	5.508558	

```
raceId         903       904       905       906       907       908  \
driverId
292       5.013214  5.092618  4.505289  3.001609  5.132125  4.159788
475       4.920542  3.677727  6.497154  3.170363  5.765657  1.752851
592       4.910499  4.930041  4.486564  2.971425  5.015485  3.997268
94        4.017436  4.165550  3.027780  2.074232  3.661415  4.124302
771       4.993167  4.993714  4.611412  3.021407  5.138568  4.013924

raceId         909       910       911       912       913       914  \
driverId
292       4.072425  4.939065  4.355166  5.330346  4.903605  4.552302
475       1.990722  7.102016  5.384028  4.014778  4.943084  2.170901
592       3.921336  4.908238  4.269780  5.153606  4.819170  4.389873
94        3.173776  3.708144  2.825477  4.413714  4.643117  4.003112
771       3.962006  5.029640  4.374748  5.230146  4.875093  4.420078

raceId         915       916       917        918       926       927  \
driverId
292       4.797651  4.962828  5.148323  10.281156  4.942573  4.746946
475       4.016283  3.920014  6.896573   8.946795  3.885317  2.756292
592       4.662780  4.818231  5.112821   9.978376  4.815272  4.586375
94        3.863893  4.062742  3.861368   9.466413  4.345481  4.195550
771       4.739521  4.892531  5.226062  10.124961  4.868404  4.611708

raceId         928       929       930       931       932       933  \
driverId
292       4.863879  5.012923  4.910015  4.859207  4.793040  4.571399
475       3.036107  3.808423  3.142811  4.636573  2.725427  2.504414
592       4.701646  4.849557  4.735959  4.741528  4.610642  4.392136
94        4.158940  3.987257  3.837459  3.625818  3.971468  3.722617
771       4.743326  4.916092  4.791730  4.817688  4.658854  4.440439

raceId         934       936       937       938       939       940  \
driverId
292       4.640436  3.381563  4.810984  4.206961  3.907836  4.728622
475       2.131418  2.321305  5.181304  1.895546  3.489278  2.630422
592       4.463399  3.296436  4.692945  4.057087  3.805852  4.553669
94        4.129958  3.324662  3.682993  4.203608  2.981144  3.968334
771       4.490883  3.300140  4.791900  4.064831  3.851607  4.591989

raceId         941       942       943       944       945       948  \
driverId
292       4.181935  5.242695  4.526668  4.706463  5.309690  4.530112
475       3.618072  4.926786  3.045174  3.096467  5.884274  2.514663
592       4.092682  5.103518  4.345465  4.540925  5.192474  4.364348
94        4.106055  4.337644  3.307057  3.629375  3.764735  3.675154
```

```
771       4.125679  5.183842  4.424290  4.593785  5.300490  4.407140

raceId         949       950       951       952       953       954  \
driverId
292       5.154905  4.550525  5.102907  4.318164  4.669935  4.676228
475       6.605487  4.729767  5.758007  6.190990  4.613300  1.646951
592       5.044964  4.447494  4.982923  4.247230  4.569607  4.467235
94        3.332577  3.255224  3.561948  3.296069  4.042791  4.287447
771       5.179289  4.525481  5.099949  4.345191  4.640897  4.484318

raceId         955       956       957       958       959       960  \
driverId
292       5.043574  5.388107  5.565981  5.374130  4.970323  4.731433
475       5.092594  6.541330  6.551049  4.767125  3.836821  4.527730
592       4.909220  5.266017  5.427152  5.219949  4.802793  4.598749
94        3.555658  4.089816  4.136740  4.305361  4.142443  3.337081
771       5.003168  5.387973  5.556399  5.302048  4.868512  4.693705

raceId         961       962       963       964       965       966  \
driverId
292       5.179558  5.301273  4.494734  5.334005  5.160578  5.096715
475       4.044001  6.075172  6.661128  5.583512  5.009562  3.645174
592       5.006523  5.180100  4.404094  5.184551  5.044536  4.913318
94        3.883938  3.676373  2.727350  3.841445  4.222794  4.103388
771       5.082187  5.295994  4.546464  5.290826  5.127439  4.981393

raceId         967       968       969       970       971       972  \
driverId
292       5.398007  5.177021  5.174865  5.929523  5.703780  5.338914
475       5.633951  4.026946  3.329126  4.544404  3.196604  3.422370
592       5.256401  5.010869  4.999416  5.731411  5.493987  5.124778
94        4.254414  4.258694  4.540556  4.822285  4.571056  4.032561
771       5.360249  5.076153  5.037330  5.807349  5.542974  5.192268

raceId         973       974       975       976       977       978  \
driverId
292       5.583385  5.972362  6.170794  5.129216  5.753632  6.241169
475       4.786429  5.655995  3.588338  3.032914  2.923407  4.552845
592       5.432800  5.803520  5.917040  4.919562  5.502432  5.997118
94        4.514994  4.178085  4.558427  3.561929  4.044129  4.554219
771       5.512489  5.902950  6.003926  4.997813  5.574248  6.101706

raceId         979       980       981       982       983       984  \
driverId
292       5.359309  6.152629  5.941341  6.391460  4.630511  5.946008
475       5.126239  3.859868  3.026801  6.459013  3.111733  5.306890
592       5.231492  5.931978  5.695748  6.199598  4.442770  5.731233
```

```
94          4.416818  4.757209  4.646140  4.304099  3.993819  4.314396
771         5.298724  6.002438  5.759606  6.354342  4.498776  5.853341

raceId         985       986       987       988       989       990   \
driverId
292         5.761383  4.202079  5.414925  5.314575  5.264737  3.989388
475         4.891623  4.303260  3.434525  2.793531  4.555766  0.636178
592         5.590863  4.066516  5.219786  5.084567  5.143404  3.817358
94          4.787814  3.170224  4.361022  4.197825  4.628411  3.861969
771         5.665354  4.143700  5.269262  5.141200  5.190667  3.804795

raceId         991       992       993       994       995       996   \
driverId
292         4.527140  6.203791  4.784617  4.839548  4.391006  5.126258
475         3.245531  8.909608  2.083616  2.502180  1.448171  4.303281
592         4.363901  6.148328  4.577511  4.661573  4.195954  4.969722
94          3.581120  4.759688  4.427993  4.033172  3.965104  4.455574
771         4.426609  6.292718  4.609046  4.695978  4.202992  5.033789

raceId         997       998       999       1000      1001      1002  \
driverId
292         3.374429  5.089169  4.497207  5.438659  4.175951  5.242971
475         5.522294  2.675325  3.075486  3.560796  1.509087  4.119406
592         3.357333  4.916285  4.326503  5.270382  4.001163  5.079654
94          2.873150  4.411869  3.911006  4.695957  4.155340  4.565483
771         3.423413  4.940353  4.381843  5.315010  4.001330  5.139574

raceId         1003      1004      1005      1006      1007      1008  \
driverId
292         5.278235  5.299769  4.905080  4.841699  4.346501  5.236200
475         3.777147  2.265228  2.421977  5.552538  3.542834  3.858202
592         5.102453  5.068129  4.693393  4.741554  4.204576  5.047813
94          4.763213  4.525158  4.363706  4.054809  3.814332  4.387327
771         5.152543  5.105467  4.734635  4.819607  4.247453  5.113672

raceId         1009      1010      1011      1012      1013      1014  \
driverId
292         5.307640  3.783604  4.950833  4.824777  4.557535  4.746943
475         3.247506 -0.476399  2.114973  1.512155  1.659196  2.070927
592         5.110216  3.529869  4.713940  4.589290  4.332622  4.532035
94          4.685296  3.329500  3.999824  4.255176  3.835641  4.244436
771         5.158924  3.531668  4.764030  4.618338  4.369754  4.569412

raceId         1015      1016      1017      1018      1019      1020  \
driverId
292         4.848702  4.905191  5.231965  3.884760  5.288893  2.935215
475         2.768588  1.507622  3.367158  2.208721  4.353388  4.168749
```

```
592       4.678157  4.667806  5.024386  3.698595  5.101200  2.913258
94        4.685269  4.173507  4.250602  3.134346  4.137308  3.033991
771       4.705882  4.698365  5.092610  3.745459  5.198507  2.950025

raceId        1021      1022      1023      1024      1025      1026  \
driverId
292       5.493053  4.736176  4.329003  4.022102  5.093007  4.543330
475       5.890704  2.362995  1.828693  0.944706  3.889445  1.563127
592       5.367497  4.521445  4.093988  3.815753  4.893852  4.338935
94        4.728528  3.431165  2.800649  3.411782  4.001135  3.874528
771       5.457620  4.586496  4.172257  3.827843  4.986121  4.368161

raceId        1027      1028      1029      1030      1031      1032  \
driverId
292       4.884580  4.068379  3.983497  4.879147  4.735257  4.437241
475       1.388766  0.750065  8.233873  2.771714  4.087034  2.642213
592       4.644374  3.817075  4.011624  4.665338  4.565038  4.250825
94        4.214566  3.178365  3.315365  4.134844  3.140410  3.966925
771       4.672905  3.861072  4.139379  4.722619  4.669293  4.304137

raceId        1033      1034      1035      1036      1037      1038  \
driverId
292       4.383680  5.361790  4.082274  4.770367  4.234975  4.634499
475       2.180446  6.043121  2.717092  4.300152  1.330849  8.001826
592       4.196794  5.221939  3.894212  4.629239  4.018124  4.644420
94        4.227679  4.217258  3.464975  4.436337  3.842034  3.629550
771       4.229507  5.340658  3.958267  4.697819  4.051561  4.777111

raceId        1039      1040      1041      1042      1043      1044  \
driverId
292       4.830504  3.677705  5.441544  5.174870  5.340545  5.990420
475       2.369550  0.906206  7.875005  4.507732  4.214682  7.832154
592       4.619986  3.452627  5.371430  4.998340  5.155070  5.905798
94        3.918329  3.032685  4.539975  4.272651  4.046146  4.883768
771       4.679540  3.494191  5.509310  5.092461  5.252454  6.034471

raceId        1045       1046      1047      1051      1052      1053  \
driverId
292       4.939573   6.743311  3.632632  5.072974  5.009917  4.835261
475       5.649796  10.444471  2.071713  7.872115  3.939094  6.373572
592       4.833250   6.706757  3.464225  5.031980  4.819248  4.736590
94        4.370144   5.578995  3.328892  4.458848  4.115292  3.831704
771       4.926319   6.886042  3.505492  5.159150  4.903991  4.853702

raceId        1054      1055      1056      1057      1058      1059  \
driverId
292       4.927751  5.124965  4.033464  4.113444  4.559246  4.236853
```

```
475         4.057408  4.331810  7.041703  6.888579  4.922272  4.011547
592         4.748084  4.935152  4.024075  4.110390  4.424146  4.100226
94          4.138455  4.137854  3.526927  2.980465  3.868052  3.855711
771         4.831624  5.031145  4.126049  4.220539  4.517285  4.168643

raceId          1060      1061      1062      1063      1064      1065  \
driverId
292         4.176986  5.705431  4.466078  1.855910  4.611681  4.622555
475         4.211990  5.150601  7.856747  1.997404  5.078616  5.961772
592         4.038912  5.526252  4.482618  1.808353  4.483160  4.524502
94          3.469982  4.088560  3.758176  1.687007  3.951839  2.722548
771         4.120998  5.644862  4.607192  1.836905  4.576755  4.657294

raceId          1066      1067      1069      1070      1071      1072  \
driverId
292         5.319452  4.013170  4.737870  4.694671  5.085431  4.681177
475         6.369946  3.157280  5.291818  6.917386  5.182900  2.936633
592         5.210957  3.841858  4.597150  4.628616  4.932118  4.485556
94          4.534932  3.116039  3.787964  3.979140  4.239374  4.100198
771         5.320831  3.920013  4.704751  4.743273  5.035827  4.548384

raceId          1073      1074       1075      1076      1077      1078  \
driverId
292         4.242235  4.601846   4.080002  5.402043  3.775140  4.704262
475         5.758348  8.508635  10.135996  8.262679  6.336423  8.651711
592         4.170950  4.564846   4.113678  5.319819  3.723916  4.665406
94          3.850570  2.462054   2.363897  3.116875  2.699250  3.258028
771         4.263025  4.760013   4.312931  5.497584  3.835108  4.829686

raceId          1079       1080      1081      1082       1083      1084  \
driverId
292         4.215140   5.174957  4.232122  4.702744   6.055144  4.803761
475         7.985680  11.606665  7.295033  7.998901  11.890079  6.969261
592         4.202392   5.223287  4.214588  4.661446   6.083820  4.711550
94          3.415344   3.414628  3.704865  3.656251   4.283792  3.177568
771         4.336425   5.430063  4.320911  4.802563   6.293263  4.851638

raceId          1085      1086       1087      1088      1089       1091  \
driverId
292         4.857583  5.195121   4.347790  4.878132  4.944748   5.235833
475         9.191692  9.293601  10.583617  9.371596  9.516260  10.177878
592         4.859822  5.171393   4.415177  4.864261  4.925643   5.236353
94          4.046635  4.067736   3.207936  3.509510  3.371551   3.158699
771         5.006632  5.328586   4.588376  5.029338  5.099939   5.430331

raceId          1092      1093      1094       1095      1096      1098  \
driverId
```

```
292       4.520666  5.128280  4.916553   5.446645  4.570879  4.502494
475       7.411608  8.011154  8.263701  10.574805  9.857041  9.620101
592       4.472347  5.062074  4.867751   5.461507  4.580862  4.537496
94        3.493861  3.906770  3.828007   3.854265  2.887898  3.715464
771       4.598173  5.201579  5.014887   5.648568  4.761796  4.694139

raceId         1099      1100       1101       1102      1104      1105  \
driverId
292        4.500055  5.362659   4.218137   4.551798  5.119380  4.859027
475       11.004301  7.050320  11.214280  10.758865  8.451535  8.642106
592        4.564224  5.303189   4.280338   4.610210  5.103507  4.849109
94         3.227301  5.001637   2.513150   3.461080  4.419638  4.211103
771        4.762487  5.383864   4.507351   4.792238  5.221024  4.980232

raceId        1106       1107      1108      1109      1110      1111  \
driverId
292       4.917150   4.526744  5.989987  5.531545  4.631442  5.030133
475       9.411629  10.249828  6.662828  7.751275  8.909379  9.484383
592       4.911030   4.552428  5.889111  5.463798  4.612943  5.060373
94        3.824899   2.978980  5.404686  4.574111  3.382221  4.452251
771       5.078062   4.740388  5.949709  5.567473  4.775805  5.188441

raceId        1112      1113      1114      1115      1116      1117  \
driverId
292       4.160510  5.819708  5.930528  5.216342  4.373902  5.398463
475       9.775973  9.710529  7.524334  6.407939  8.832719  7.556349
592       4.181224  5.793228  5.842614  5.137849  4.406352  5.309766
94        2.723483  4.225948  4.824256  4.280013  3.570896  4.189569
771       4.372702  5.960210  5.936492  5.204977  4.528415  5.437741

raceId        1118      1119      1120       1121       1122       1123  \
driverId
292       5.068467  3.682620  4.467543   4.522025   4.994594   4.654723
475       9.137365  8.847159  8.526721  10.336165  11.685788  11.166448
592       5.091167  3.692205  4.452310   4.549350   5.039922   4.722317
94        4.443330  2.210577  3.039283   3.015418   3.164717   2.416491
771       5.204430  3.877372  4.597593   4.739480   5.254951   4.939580

raceId         1124      1125      1126      1127      1128      1129  \
driverId
292        4.550395  4.850347  4.952109  5.400541  5.231192  5.984355
475       10.922088  9.934895  8.927100  8.000293  8.894958  6.343543
592        4.601078  4.868837  4.935021  5.337659  5.195242  5.890255
94         3.028672  3.455910  3.501457  4.073471  3.158180  5.571060
771        4.796180  5.030726  5.085600  5.457663  5.361485  5.931504

raceId        1130      1131      1132      1133      1134      1135  \
```

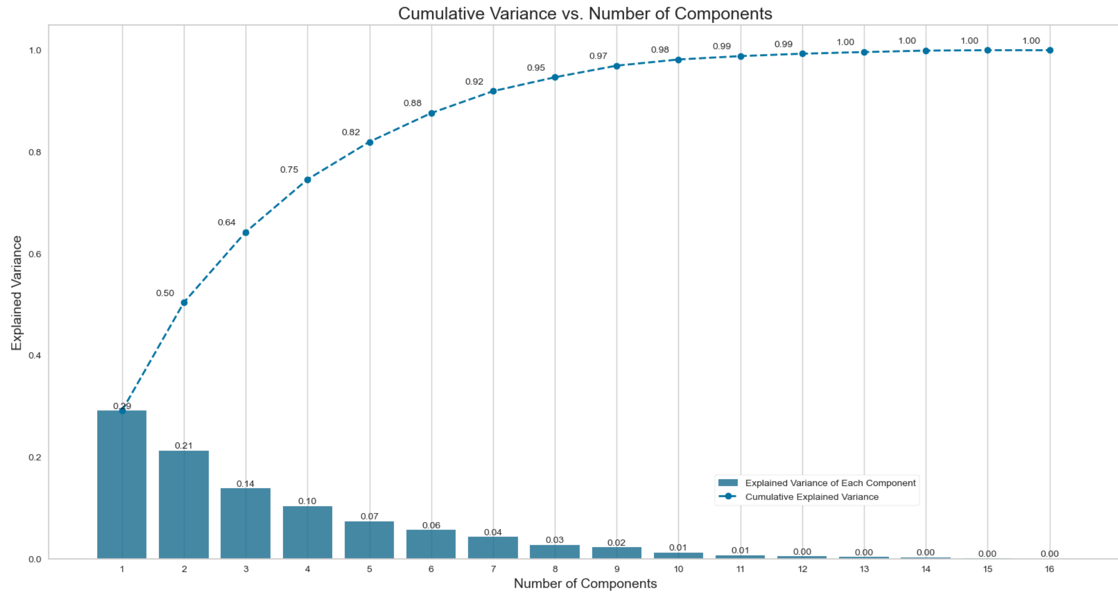
```
driverId
292       5.698927  5.222522  6.413246  6.285454  6.312309  5.489009
475       7.668575  8.141124  6.227240  7.185662  7.534684  8.607872
592       5.620450  5.202777  6.289502  6.177029  6.193891  5.448687
94        4.690892  4.130893  5.904610  4.860134  4.894295  3.971627
771       5.723096  5.316715  6.345206  6.273535  6.317169  5.577111

raceId        1136      1137      1138       1139       1140      1141  \
driverId
292       5.494310  5.426676  5.982681   4.575898   5.432527  3.903548
475       8.608127  6.977996  7.475919  10.040394  10.119826  5.794740
592       5.434961  5.357209  5.907831   4.587914   5.432271  3.861583
94        3.436574  3.916838  4.850823   2.398607   3.596667  3.257308
771       5.595635  5.450160  5.989634   4.789746   5.613436  3.935638

raceId        1142      1143      1144
driverId
292       5.530615  4.568390  5.573679
475       9.014047  8.250928  9.359590
592       5.497543  4.551744  5.555964
94        4.032199  3.191032  3.972311
771       5.647628  4.686325  5.708422
```

We used a rank-5 iterative SVD to fill all of approximate 942K missing entries in the drivers×races points matrix. After 100 iterations (relative change is 3×10^{3}), 0 NaNs remain and the imputed scores for drivers (e.g. IDs $715, 606, 121, 209, 381$) vary smoothly and plausibly.

2.1 SVD

[164]:
```python
np.random.seed(42)

U, S, VT = np.linalg.svd(data_scaled_df, full_matrices=False)

explained_variance = (S ** 2) / np.sum(S ** 2) # Explained variance
svd_projection = U[:, :2] @ np.diag(S[:2]) # 2D SVD Projection

#sns.set(rc={'axes.facecolor': '#fcf0dc'}, style='darkgrid')
cumulative_explained_variance = np.cumsum(explained_variance)

# Plot the cumulative explained variance against the number of components
plt.figure(figsize=(20, 10))

# Bar chart for the explained variance of each component
barplot = sns.barplot(x=list(range(1, len(cumulative_explained_variance) + 1)),
                      y=explained_variance,
                      alpha=0.8)

# Line plot for the cumulative explained variance
```

```
lineplot, = plt.plot(range(0, len(cumulative_explained_variance)),␣
↪cumulative_explained_variance,
                     marker='o', linestyle='--',  linewidth=2)

# Set labels and title
plt.xlabel('Number of Components', fontsize=14)
plt.ylabel('Explained Variance', fontsize=14)
plt.title('Cumulative Variance vs. Number of Components', fontsize=18)

# Customize ticks and legend
plt.xticks(range(0, len(cumulative_explained_variance)))
plt.legend(handles=[barplot.patches[0], lineplot],
           labels=['Explained Variance of Each Component', 'Cumulative␣
↪Explained Variance'],
           loc=(0.62, 0.1),
           frameon=True,
           framealpha=1.0,
           )

# Display the variance values for both graphs on the plots
x_offset = -0.3
y_offset = 0.01
for i, (ev_ratio, cum_ev_ratio) in enumerate(zip(explained_variance,␣
↪cumulative_explained_variance)):
    plt.text(i, ev_ratio, f"{ev_ratio:.2f}", ha="center", va="bottom",␣
↪fontsize=10)
    if i > 0:
        plt.text(i + x_offset, cum_ev_ratio + y_offset, f"{cum_ev_ratio:.2f}",␣
↪ha="center", va="bottom", fontsize=10)

plt.grid(axis='both')
plt.show()
```



The plot and the cumulative explained variance values indicate how much of the total variance in the dataset is captured by each component, as well as the cumulative variance explained by the first n components.

Here, we can observe that:

The first component explains approximately 31% of the variance.
The first two components together explain about 54% of the variance.
The first six components explain approximately 88% of the variance, and so on.

Choosing optimal number of components as 6 after which adding another component doesn't significantly increase in the cummulative variance, often referred to as the "elbow point" in the curve.

From the plot, we can see that the increase in cumulative variance starts to slow down after the 6th component (which captures about 88% of the total variance) and also adding another component has no change in the variance of the individua component.

2.2 PCA

[165]:
```
# performing PCA with 9 components
pca = PCA(n_components=6)
F1_data_pca = pca.fit_transform(data_scaled_df)

# Creating a new dataframe from the PCA dataframe, with columns labeled PC1,␣
↪PC2, etc.
F1_data_pca = pd.DataFrame(F1_data_pca, columns=['PC'+str(i+1) for i in␣
↪range(pca.n_components_)])

id_index = final_data[["raceId", "driverId"]].reset_index(drop=True)
```

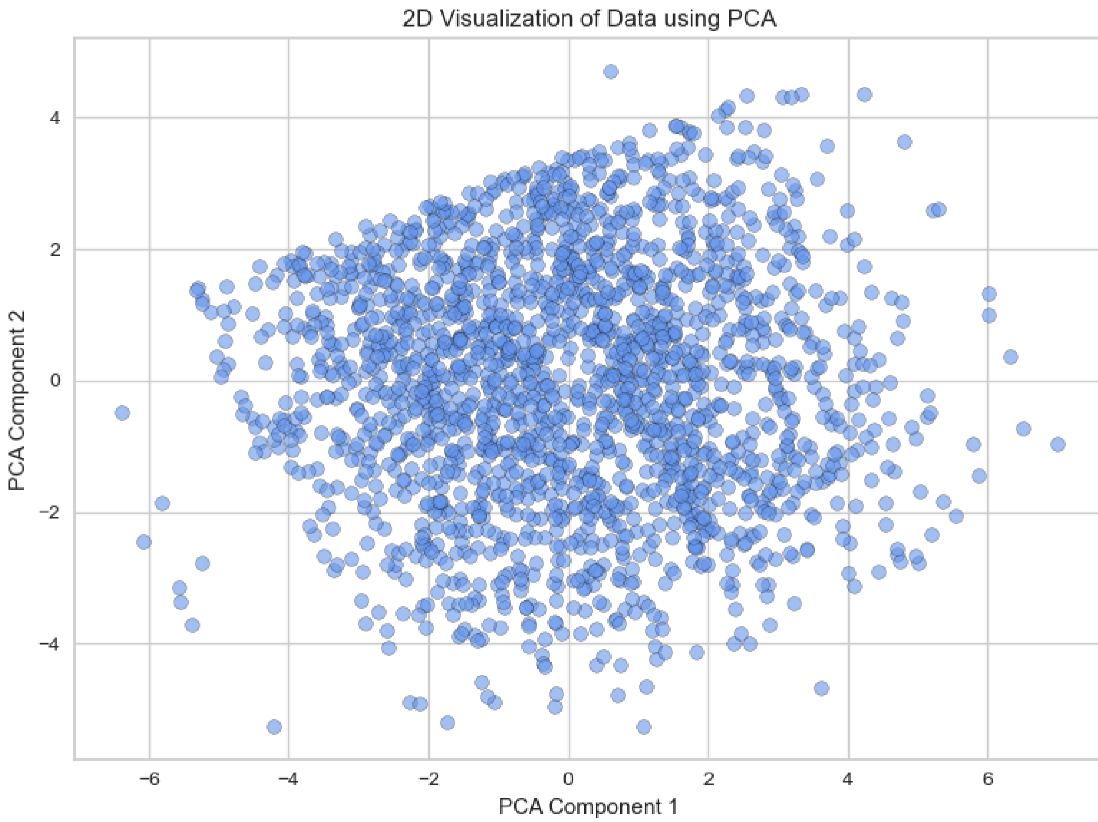
```
pca_df = pd.concat([id_index, F1_data_pca], axis=1).set_index(["raceId",␣
↪"driverId"])
pca_df = pca_df.reset_index()
pca_df.head()
```

[165]:

```
   raceId  driverId       PC1       PC2       PC3       PC4       PC5  \
0     841        20 -1.454654  2.467678 -0.395891 -0.405817  0.197826
1     841         1 -0.852697  1.307286 -0.500208 -0.520739  0.199261
2     841       808 -0.563475 -0.022435 -0.643267  1.149230 -0.113728
3     841         4 -0.646255  0.312125 -0.114757  0.235873  0.206241
4     841        17 -0.756625  0.405742  0.096841 -1.191356  0.212870

        PC6
0 -0.369316
1 -0.652279
2  0.262184
3  1.254385
4  1.058389
```

[166]:

```
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA

# Reduce to 2 dimensions for visualization
pca_2d = PCA(n_components=2)
X_pca_2d = pca_2d.fit_transform(data_scaled_df)

# Plot the 2D projection
plt.figure(figsize=(8, 6))
plt.scatter(X_pca_2d[:, 0], X_pca_2d[:, 1], alpha=0.6, c='cornflowerblue',␣
↪edgecolors='k')
plt.xlabel("PCA Component 1")
plt.ylabel("PCA Component 2")
plt.title("2D Visualization of Data using PCA")
plt.grid(True)
plt.tight_layout()
plt.show()
```



There is no clear seperation between the data when we visualize the 2 components of PCA. Next we try to perform K-mean on this data to visualize the clusters.

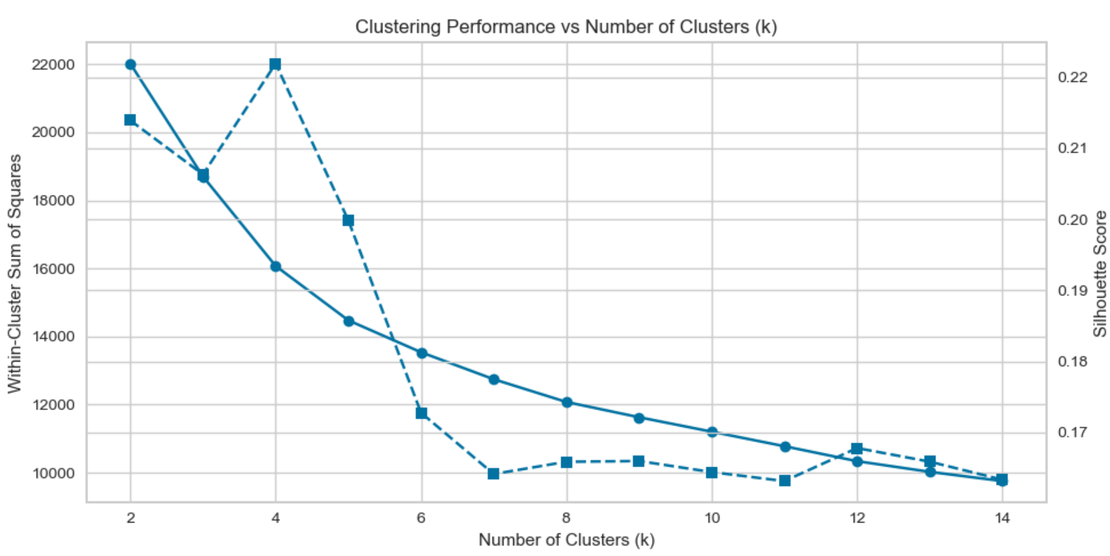
[167]:
```
def highlight_top3(column):
    top3 = column.abs().nlargest(3).index
    return ['background-color:  #ffeacc' if i in top3 else '' for i in column.
 ↪index]

# Create the PCA component DataFrame and apply the highlighting function
pc_df = pd.DataFrame(pca.components_.T, columns=['PC{}'.format(i+1) for i in␣
 ↪range(pca.n_components_)],
                     index=data_scaled_df.columns)

pc_df.style.apply(highlight_top3, axis=0)
```

[167]: <pandas.io.formats.style.Styler at 0x1ca26b95ee0>

2.3 K Means

2.3.1 Determining the optimal number of clusters

[168]:
```
ks = range(2, 15)
wss = []
silhouette_scores = []

for k in ks:
    kmeans = KMeans(n_clusters=k, n_init=20, random_state=42)
    labels = kmeans.fit_predict(F1_data_pca)
    wss.append(kmeans.inertia_)
    sil_score = silhouette_score(F1_data_pca, labels)
    silhouette_scores.append(sil_score)

# Plot
fig, ax1 = plt.subplots(figsize=(10, 5))

ax1.set_xlabel('Number of Clusters (k)')
ax1.set_ylabel('Within-Cluster Sum of Squares')
ax1.plot(ks, wss, marker='o', label='WSS')
ax1.tick_params(axis='y')

ax2 = ax1.twinx()
ax2.set_ylabel('Silhouette Score')
ax2.plot(ks, silhouette_scores, marker='s', linestyle='--', label='Silhouette')
ax2.tick_params(axis='y')

plt.title('Clustering Performance vs Number of Clusters (k)')
fig.tight_layout()
plt.grid(True)
plt.show()
```



[169]:
```python
kmeans = KMeans(n_clusters=4, init='k-means++', n_init=9, max_iter=100,␣
↪random_state=0)
kmeans.fit(F1_data_pca)

# Optional: get cluster frequencies
cluster_frequencies = Counter(kmeans.labels_)

# Optional: sort clusters by size (optional relabeling)
sorted_clusters = [c for c, _ in cluster_frequencies.most_common()]
relabel = {old: new for new, old in enumerate(sorted_clusters)}
new_labels = np.array([relabel[l] for l in kmeans.labels_])

# Add cluster labels
F1_data_pca["cluster"] = new_labels
data_scaled_df["cluster"] = new_labels
```

[170]:
```python
data_scaled_df.head()
```

[170]:
```
       grid  finalPosition    points  Total_time_ms  fastestLap  \
0 -1.290033      -1.235402  1.737068      -0.439229   -0.391093
1 -0.964747      -0.912358  0.794654      -0.420336   -0.635854
2  0.336397      -0.589315  0.390762      -0.413335    0.506365
3  0.011111      -0.266271 -0.013130      -0.412308    0.016842
4 -0.639461       0.056772 -0.282391      -0.406886    0.098430

   fastesLapRank  fastestLapSpeed  qualifyingPosition  total_pit_stops  \
0      -0.513603         0.236956           -1.414751        -0.027390
1       0.465612         0.184532           -1.046952        -0.027390
2       0.220808         0.212356            0.424243        -0.027390
3      -1.003211         0.277152            0.056444         1.016529
4      -0.758407         0.264354           -0.679154         1.016529

   average_pit_duration_ms  avg_lap_time_ms  position_change  \
0                -0.269378        -0.389816        -0.048314
1                -0.269930        -0.369424        -0.048314
2                -0.260102        -0.361867         0.975400
3                -0.265566        -0.360759         0.292924
4                -0.265547        -0.354907        -0.730789

   fastestLapTime_ms     q1_ms     q2_ms     q3_ms  cluster
0          -0.065757 -0.250179 -0.293561 -0.325067        0
1          -0.025557 -0.243077 -0.252019 -0.262457        0
2          -0.046940 -0.230246 -0.170826 -0.186809        0
3          -0.096292 -0.217011 -0.198795 -0.208779        0
```

```
4          -0.086627 -0.201436 -0.246836 -0.255375       0
```

[171]:

```
colors = ['#e8000b', '#1ac938', '#023eff', '#000000']
cluster_0 = F1_data_pca[F1_data_pca['cluster'] == 0]
cluster_1 = F1_data_pca[F1_data_pca['cluster'] == 1]
cluster_2 = F1_data_pca[F1_data_pca['cluster'] == 2]
#cluster_3 = F1_data_pca[F1_data_pca['cluster'] == 3]
# Create a 3D scatter plot
fig = go.Figure()

# Add data points for each cluster separately and specify the color
fig.add_trace(go.Scatter3d(x=cluster_0['PC1'], y=cluster_0['PC2'],␣
 ↪z=cluster_0['PC3'],
                           mode='markers', marker=dict(color=colors[0], size=5,␣
 ↪opacity=0.4), name='Cluster 0'))
fig.add_trace(go.Scatter3d(x=cluster_1['PC1'], y=cluster_1['PC2'],␣
 ↪z=cluster_1['PC3'],
                           mode='markers', marker=dict(color=colors[1], size=5,␣
 ↪opacity=0.4), name='Cluster 1'))
fig.add_trace(go.Scatter3d(x=cluster_2['PC1'], y=cluster_2['PC2'],␣
 ↪z=cluster_2['PC3'],
                           mode='markers', marker=dict(color=colors[2], size=5,␣
 ↪opacity=0.4), name='Cluster 2'))
#fig.add_trace(go.Scatter3d(x=cluster_3['PC1'], y=cluster_3['PC2'],␣
 ↪z=cluster_3['PC3'],
                           #mode='markers', marker=dict(color=colors[3],␣
 ↪size=5, opacity=0.4), name='Cluster 3'))

# Set the title and layout details
fig.update_layout(
    title=dict(text='3D Visualization of first 3 Clusters in PCA Space', x=0.5),
    scene=dict(
        xaxis=dict(backgroundcolor="#fcf0dc", gridcolor='white', title='PC1'),
        yaxis=dict(backgroundcolor="#fcf0dc", gridcolor='white', title='PC2'),
        zaxis=dict(backgroundcolor="#fcf0dc", gridcolor='white', title='PC3'),
    ),
    width=900,
    height=800
)

# Show the plot
fig.show()
```

[172]:

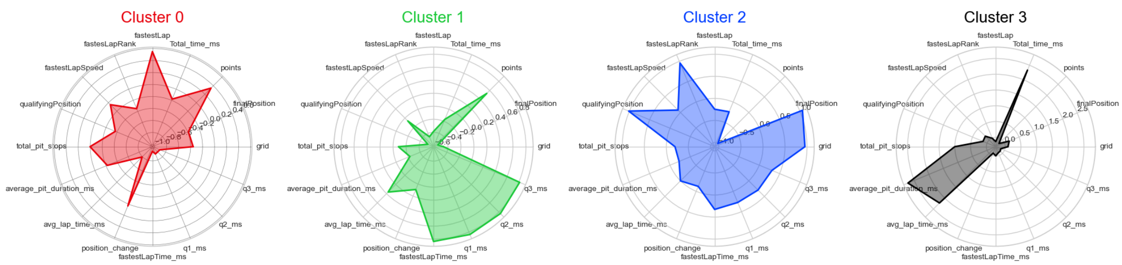
```
num_observations = len(F1_data_pca)

# Separate the features and the cluster labels
```

```python
X = F1_data_pca.drop('cluster', axis=1)
clusters = F1_data_pca['cluster']

# Compute the metrics
sil_score = silhouette_score(X, clusters)

# Create a table to display the metrics and the number of observations
table_data = [
    ["Number of Observations", num_observations],
    ["Silhouette Score", sil_score]
]

# Print the table
print(tabulate(table_data, headers=["Metric", "Value"], tablefmt='pretty'))
```

```
+------------------------+---------------------+
|         Metric         |        Value        |
+------------------------+---------------------+
| Number of Observations |        2020         |
|    Silhouette Score    | 0.22271686078883207 |
+------------------------+---------------------+
```

[173]:
```python
colors = ['#e8000b', '#1ac938', '#023eff', '#000000']

# Standardize the data (excluding the cluster column)
#scaler = StandardScaler()
df_standardized = data_scaled_df.copy()

# Create a new dataframe with standardized values and add the cluster column␣
↪back
df_standardized = pd.DataFrame(data_scaled_df, columns=data_scaled_df.columns[:
↪-1], index=data_scaled_df.index)
df_standardized['cluster'] = data_scaled_df['cluster']

# Calculate the centroids of each cluster
cluster_centroids = df_standardized.groupby('cluster').mean()

# Function to create a radar chart
def create_radar_chart(ax, angles, data, color, cluster):
    # Plot the data and fill the area
    ax.fill(angles, data, color=color, alpha=0.4)
    ax.plot(angles, data, color=color, linewidth=2, linestyle='solid')

    # Add a title
    ax.set_title(f'Cluster {cluster}', size=20, color=color, y=1.1)

# Set data
```

```
labels=np.array(cluster_centroids.columns)
num_vars = len(labels)

# Compute angle of each axis
angles = np.linspace(0, 2 * np.pi, num_vars, endpoint=False).tolist()

# The plot is circular, so we need to "complete the loop" and append the start␣
↪to the end
labels = np.concatenate((labels, [labels[0]]))
angles += angles[:1]

# Initialize the figure
fig, ax = plt.subplots(figsize=(20, 10), subplot_kw=dict(polar=True), nrows=1,␣
↪ncols=4)

# Create radar chart for each cluster
for i, color in enumerate(colors):
    data = cluster_centroids.loc[i].tolist()
    data += data[:1]  # Complete the loop
    create_radar_chart(ax[i], angles, data, color, i)

# Add input data
ax[0].set_xticks(angles[:-1])
ax[0].set_xticklabels(labels[:-1])

ax[1].set_xticks(angles[:-1])
ax[1].set_xticklabels(labels[:-1])

ax[2].set_xticks(angles[:-1])
ax[2].set_xticklabels(labels[:-1])

ax[3].set_xticks(angles[:-1])
ax[3].set_xticklabels(labels[:-1])

# Add a grid
ax[0].grid(color='grey', linewidth=0.5)

# Display the plot
plt.tight_layout()
plt.show()
```



Interpretation from radar chart
Cluster 0 (Red)
High on: fastestLap, points, positionOrder(final position)
Low on: position_change (negative) and qualifyingPosition
Interpretation: These drivers tend to finish at the top with fast lap times and strong race-day performance, even if they lose positions relative to their start. Possibly consistent, fast finishers.

Cluster 1 (Green)
High on: q1_ms, q2_ms, q3_ms (longer times), position_change

Low on: race outcomes like points, positionOrder, grid

Interpretation: Drivers in this group qualify poorly (slow qualifying laps) and often start at the back — but show strong racecraft, gaining positions during the race. Possibly skilled midfielders or rookies.

Cluster 2 (Blue)
High on: fastestLapSpeed, qualifyingPosition, positionOrder

Low on: points, pit_stops, avg_lap_time_ms

Interpretation: Drivers in this cluster are excellent qualifiers, with strong lap speeds, but may not convert those into high points. Possibly inconsistent performers or drivers in weaker cars.

Cluster 3 (Black)
High on: average_pit_duration_ms, total_pit_stops, Total_time_ms

Low on: everything else

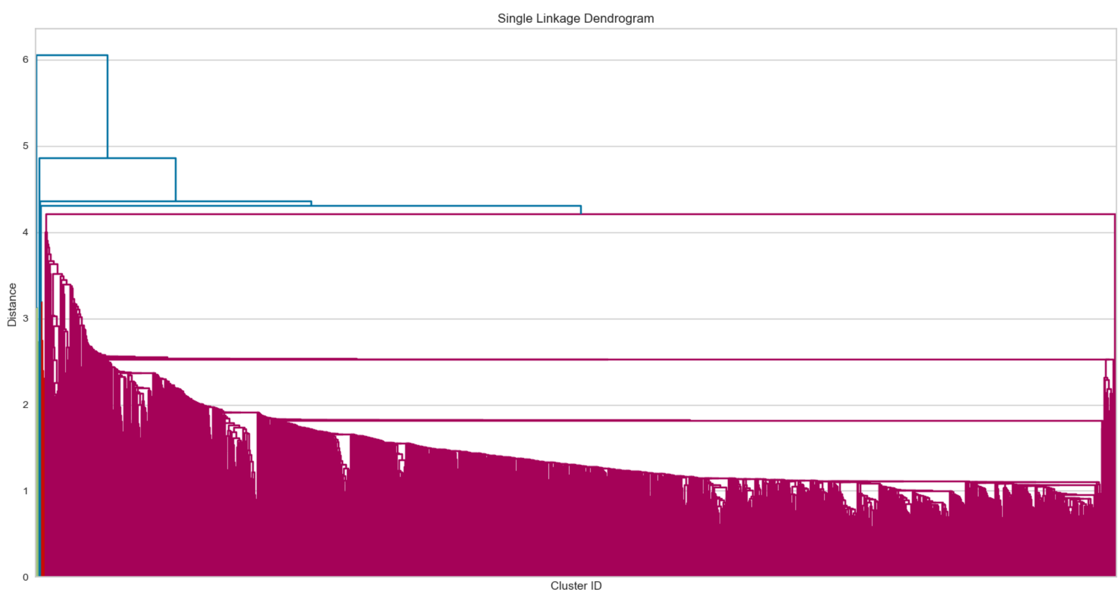
Interpretation: Likely poor performers or drivers impacted by long pit stops, technical issues, or penalties. These could be backmarkers or those who don't finish well

[ ]:

3 Hierarchical Clustering

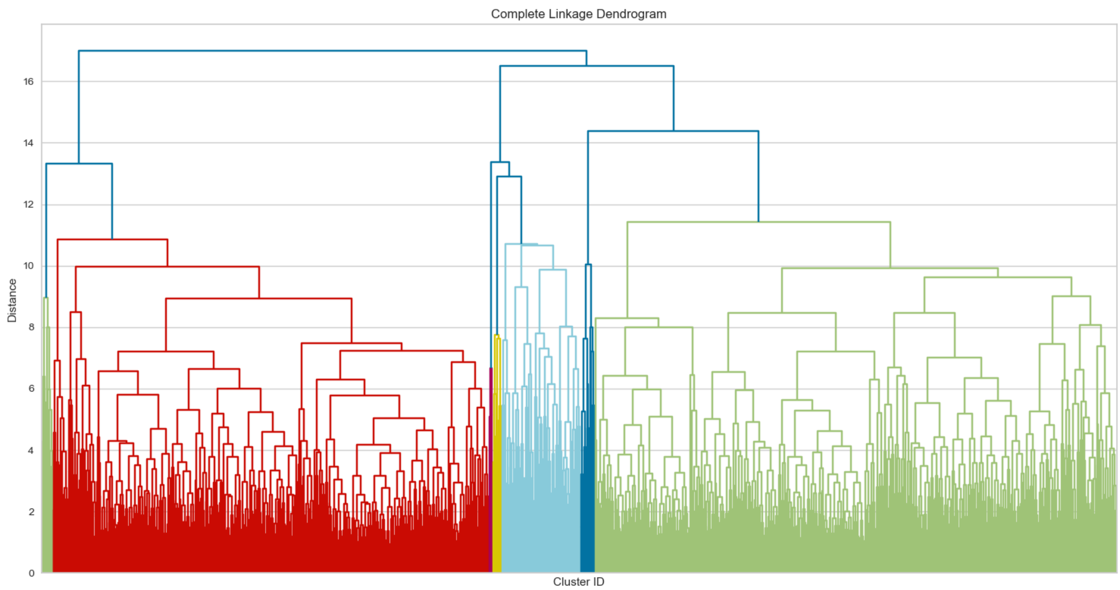
[174]:
```
linkage_methods = ['single', 'complete', 'average', 'centroid']
cluster_numbers  = [2, 3, 4, 5, 6]
```

[175]:
```python
# Looping through each types of linkage
for method in linkage_methods:

    Z = linkage(X_scaled, method=method, metric='euclidean')

    # Ploting all dendrogram
    plt.figure(figsize=(15, 8))
    dendrogram(
        Z, leaf_rotation=90, leaf_font_size=6, color_threshold=None,␣
↪no_labels=True
    )
    plt.title(f"{method.capitalize()} Linkage Dendrogram")
    plt.xlabel("Cluster ID")
    plt.ylabel("Distance")
    plt.tight_layout()
    plt.show()

    # Trying out different numbers of flat clusters
    print(f"\n {method.capitalize()} linkage:")
    for k in cluster_numbers:
        labels = fcluster(Z, t=k, criterion='maxclust')
        # bincount returns the counts for labels 1… k
        sizes = np.bincount(labels)[1:]
        print(f"k = {k:>2}  ->  cluster sizes: {sizes.tolist()}")
```

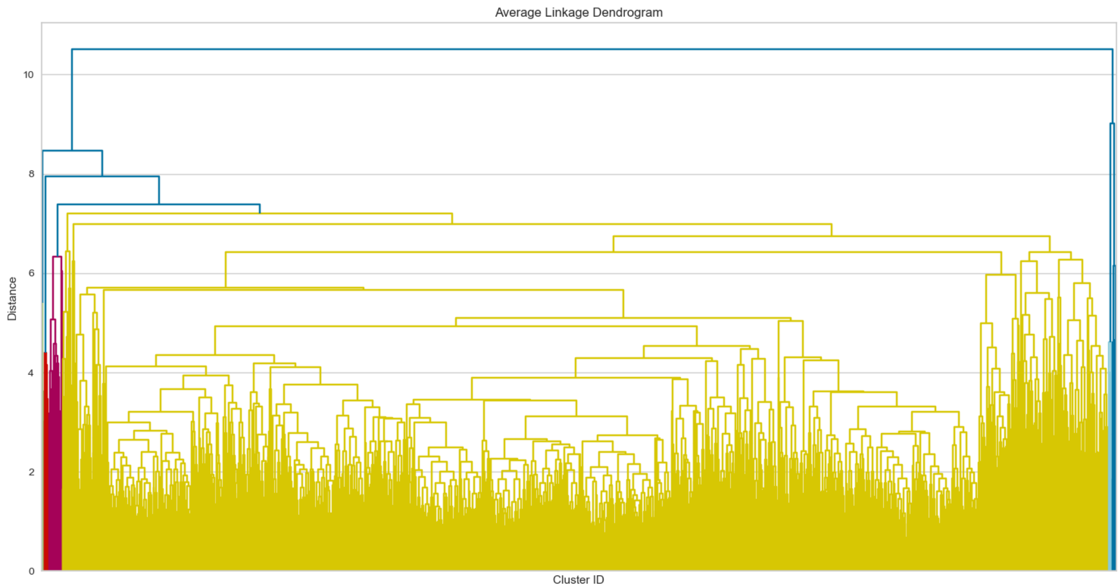



```
 Single linkage:
k =  2  ->  cluster sizes: [7, 2013]
```

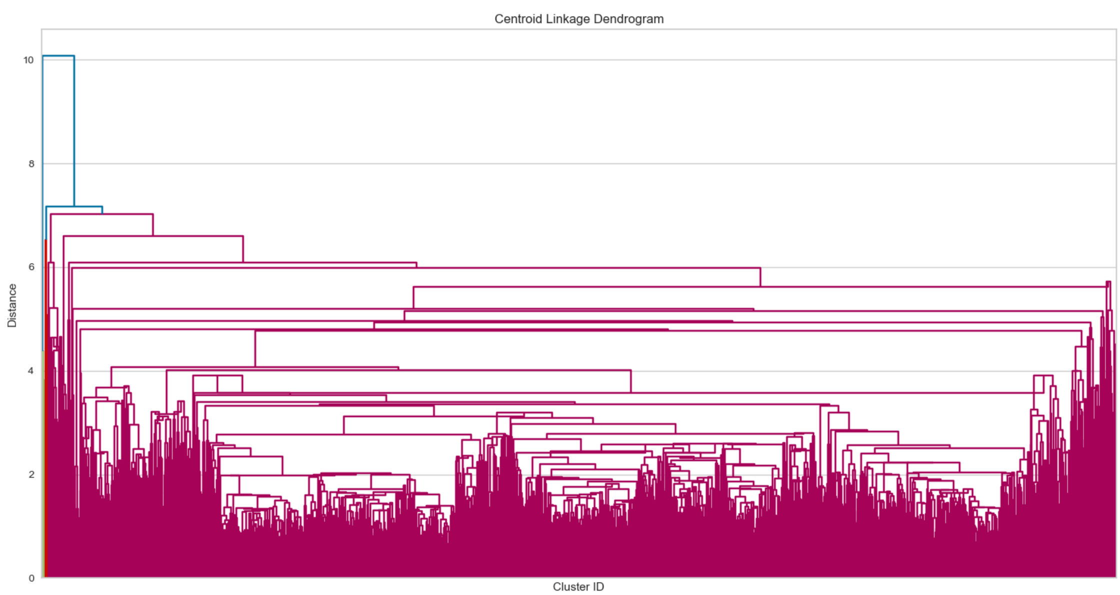
```
k =  3  ->  cluster sizes: [7, 2012, 1]
k =  4  ->  cluster sizes: [7, 2011, 1, 1]
k =  5  ->  cluster sizes: [7, 9, 2002, 1, 1]
k =  6  ->  cluster sizes: [7, 9, 1999, 3, 1, 1]
```




```
 Complete linkage:
k =  2  ->  cluster sizes: [841, 1179]
k =  3  ->  cluster sizes: [841, 172, 1007]
k =  4  ->  cluster sizes: [841, 172, 27, 980]
k =  5  ->  cluster sizes: [841, 7, 165, 27, 980]
k =  6  ->  cluster sizes: [22, 819, 7, 165, 27, 980]
```




```
Average linkage:
k =  2  ->  cluster sizes: [2005, 15]
k =  3  ->  cluster sizes: [2005, 7, 8]
k =  4  ->  cluster sizes: [5, 2000, 7, 8]
k =  5  ->  cluster sizes: [5, 9, 1991, 7, 8]
k =  6  ->  cluster sizes: [5, 9, 25, 1966, 7, 8]
```



```
 Centroid linkage:
k =  2  ->  cluster sizes: [7, 2013]
k =  3  ->  cluster sizes: [7, 6, 2007]
k =  4  ->  cluster sizes: [7, 6, 27, 1980]
k =  5  ->  cluster sizes: [7, 6, 27, 9, 1971]
k =  6  ->  cluster sizes: [7, 5, 1, 27, 9, 1971]
```

Key Observations:

— Single linkage : - Extremely imbalanced clusters at every k - One massive cluster plus very small clusters observed

— Complete linkage - It Produces more balanced mid-sized clusters at moderate values of k like 4, 5, 6 - It still isolates some true outliers but captures two or three main grouping among the bulk of the data.

— Average and Centroid linkage - Both are giving outcomes similar to the single linkage. - One dominating cluster and other very small clusters/ few outliers.

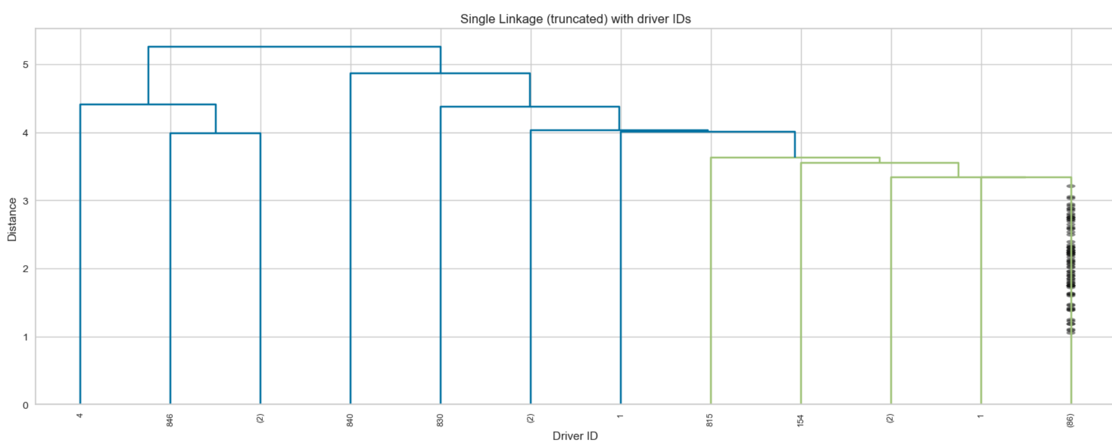
3.0.1 Scaling

[176]:
```
drop_cols = [
    "raceId","constructorId","circuitId",␣
↪"statusId","round","year","laps","hc_cluster"
]
# Here, numeric features only considered, then dropping unwanted columns
numeric_cols = final_data.select_dtypes(include="number").columns
features = final_data[numeric_cols].drop(columns=drop_cols, errors="ignore")
```

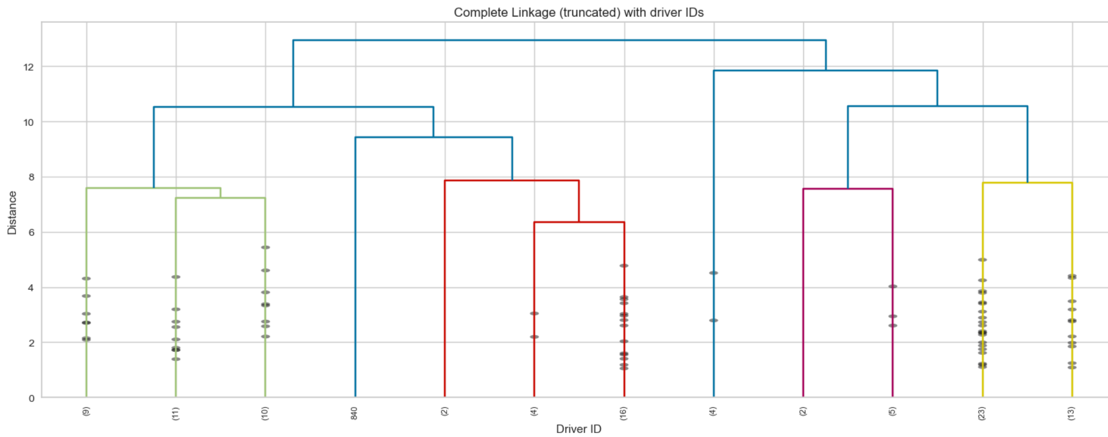
[177]:
```
# Selected random sub-sample of 100 rows for dendrogram
n_samples = 100
sub_idx   = np.random.choice(features.index, size=n_samples, replace=False)
X_sub     = features.loc[sub_idx]

driver_ids = final_data.loc[sub_idx, 'driverId'].astype(str).values

# Now, lets scale sub-sampled data
scaler     = StandardScaler()
X_scaled   = scaler.fit_transform(X_sub)
scaled_df  = pd.DataFrame(X_scaled, columns=X_sub.columns, index=X_sub.index)
```

[178]:
```
# Linkage computation and dendrograms plots
linkage_methods = ['single','complete','average','centroid']
for method in linkage_methods:
    Z = linkage(X_scaled, method=method, metric='euclidean')

    plt.figure(figsize=(15, 6))
    dendrogram(
        Z,
```

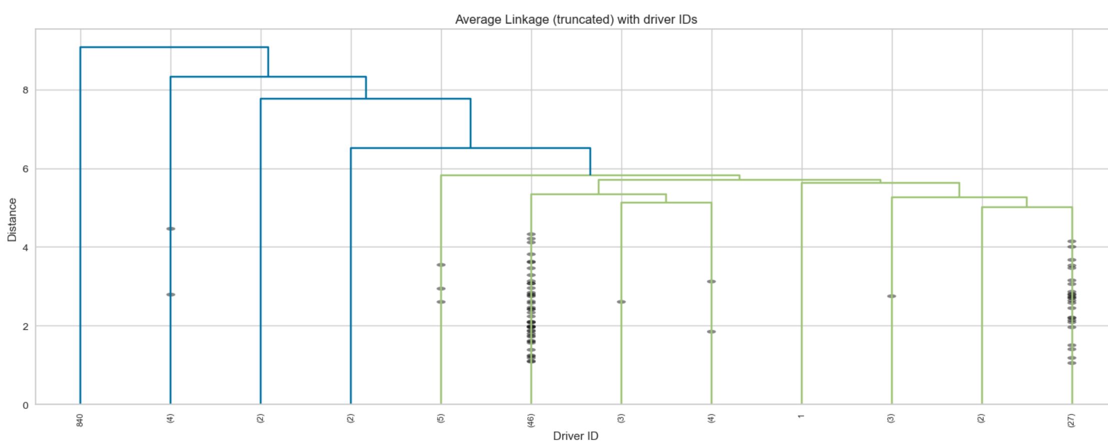
```
        truncate_mode='lastp',
        p=12,
        labels = driver_ids,
        show_contracted=True,
        leaf_rotation=90,
        leaf_font_size=8,
    )
    plt.title(f"{method.capitalize()} Linkage (truncated) with driver IDs")
    plt.xlabel("Driver ID")
    plt.ylabel("Distance")
    plt.tight_layout()
    plt.show()

    # 4) Extract flat clusters on the full linkage of the sub-sample
    print(f"\n{method.capitalize()} linkage flat clusters (n={n_samples}):")
    for k in [2, 3, 4, 5, 6]:
        labels = fcluster(Z, t=k, criterion='maxclust')
        sizes  = np.bincount(labels)[1:]
        print(f"k={k:<2} -> sizes = {sizes.tolist()}")
```



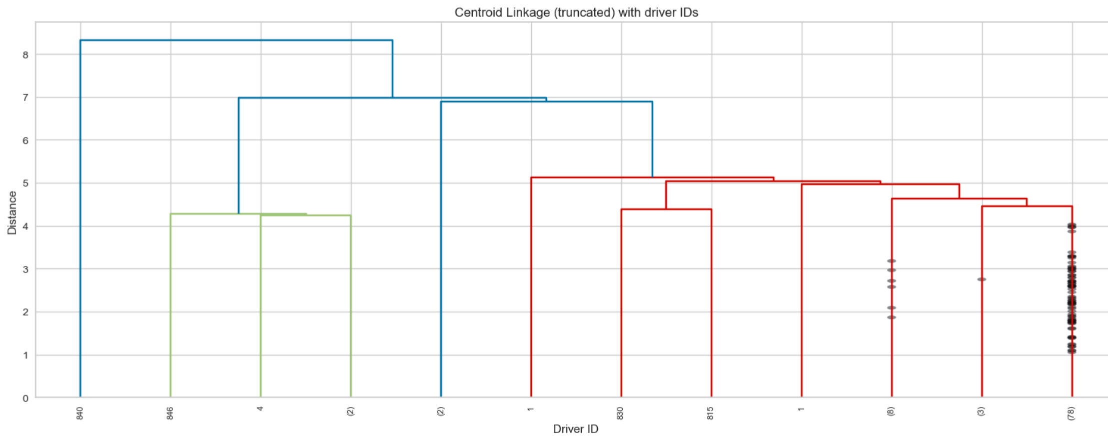

```
Single linkage flat clusters (n=100):
k=2  -> sizes = [4, 96]
k=3  -> sizes = [4, 95, 1]
k=4  -> sizes = [3, 1, 95, 1]
k=5  -> sizes = [3, 1, 94, 1, 1]
k=6  -> sizes = [3, 1, 2, 92, 1, 1]
```




```
Complete linkage flat clusters (n=100):
k=2  -> sizes = [53, 47]
k=3  -> sizes = [53, 4, 43]
k=4  -> sizes = [53, 4, 7, 36]
k=5  -> sizes = [30, 23, 4, 7, 36]
k=6  -> sizes = [30, 22, 1, 4, 7, 36]
```




```
Average linkage flat clusters (n=100):
k=2  -> sizes = [99, 1]
k=3  -> sizes = [4, 95, 1]
k=4  -> sizes = [4, 2, 93, 1]
k=5  -> sizes = [4, 2, 2, 91, 1]
k=6  -> sizes = [4, 2, 2, 5, 86, 1]
```




```
Centroid linkage flat clusters (n=100):
k=2  -> sizes = [99, 1]
k=3  -> sizes = [4, 95, 1]
k=4  -> sizes = [4, 2, 93, 1]
k=5  -> sizes = [4, 2, 92, 1, 1]
k=6  -> sizes = [4, 2, 2, 90, 1, 1]
```

We first considered 100 sub-sampled data, scaled the subsampled numeric features dropping non-numeric columns.

Effect of feature scaling on:

— Single: - Normalizing the features slightly changes the linkage distances still forms one giant cluster containing almost all points, with each outlier split off.

— Complete: - More balanced groupings as scaling yields two or three mid-sized clusters instead of a single massive one. - Outliers continue to be isolated as tiny clusters, but is now spread more evenly across clusters. - It seems to be the best linkage type for this dataset as it is properly or efficiently dividing the scaled data into clusters.

— Average: - Still produces one dominant cluster holding most points, plus a smaller secondary cluster and a few singleton outliers.

— Centroid: - Scaling makes centroid calculations more stable, but the big cluster and outliers persists.

— Difference observed after scaling data:

- Scaling balances the influence of each feature, so no single variable with a large range dominates the clustering.
- Complete linkage benefits most, producing noticeably more even cluster sizes once features are standardized.
- Single, average, and centroid linkages remain largely unchanged in pattern.

[ ]: